Nearly-Linear Time and Streaming Algorithms for Outlier-Robust PCA

Ilias Diakonikolas [1] **Daniel M. Kane** [2] **Ankit Pensia** [1] **Thanasis Pittas** [1]

Abstract

We study principal component analysis (PCA), where given a dataset in $\mathbb{R}^d$ from a distribution, the task is to find a unit vector v that approximately maximizes the variance of the distribution after being projected along v. Despite being a classical task, standard estimators fail drastically if the data contains even a small fraction of outliers, motivating the problem of robust PCA. Recent work has developed computationally-efficient algorithms for robust PCA that either take super-linear time or have sub-optimal error guarantees. Our main contribution is to develop a nearly linear time algorithm for robust PCA with near-optimal error guarantees. We also develop a single-pass streaming algorithm for robust PCA with memory usage nearly-linear in the dimension.

1. Introduction

Principal component analysis (PCA) is a central subroutine in dimension reduction and data visualization. It is used to identify directions of large variance, which are considered to be the most informative aspects of the data. In the classical setting, we observe a set S of n i.i.d. points in $\mathbb{R}^d$ from a (subgaussian) distribution with covariance $\mathbf{\Sigma}$. Then, classical estimators, for e.g., the leading eigenvector of the empirical covariance of S, output a direction v with the property that $v^\top \mathbf{\Sigma} v$ approaches $\|\mathbf{\Sigma}\|_{\mathrm{op}}$ as n increases. Importantly, these estimators are fast and have nearly linear runtime.[1]

However, access to i.i.d. samples is often an unrealistic assumption, and data may contain a small number of outliers as formalized below:

Definition 1.1 (Strong Contamination Model)**.** *Given a parameter $0 < \epsilon < 1/2$ and a class of distributions $\mathcal{D}$, the strong adversary operates as follows: The algorithm specifies a number of samples n, then the adversary draws a set of n i.i.d. samples from some $D \in \mathcal{D}$ and after inspecting them, removes up to ϵn of them and replaces them with arbitrary points. The resulting set is given as input to the learning algorithm. We call a set ϵ-corrupted if it has been generated by the above process.*

Unfortunately, outliers can skew the results of classical estimators, by artificially increasing the variance along low-variance directions, which renders the output of these estimators unreliable.[2] To address this, a robust algorithm must be able to effectively remove outliers from the data, while still being computationally efficient in high dimensions.

Recent works have developed computationally-efficient algorithms for robust PCA (Jambulapati et al., 2020; Kong et al., 2020). However, these polynomial-time algorithms either (i) achieve near-optimal error but take super-linear time (scaling with nd^2) or (ii) run in nearly linear time but achieve sub-optimal error and require additional assumptions. As modern datasets continue to expand in both sample size and dimension, there is a growing need for nearly linear time algorithms and streaming algorithms with near-optimal error guarantees. Our work addresses this need by developing robust PCA algorithms that meet these demands.

1.1. Our Results

The main result of our paper is a nearly linear time algorithm to identify a direction of large variance from a corrupted set. We state our results below for subgaussian distributions (Definition B.1), but these results hold for a more general class of "stable" distributions (Definition 2.4).

Theorem 1.2. *Let $\epsilon \in (0, c)$ for a small constant $c > 0$. Let D be a subgaussian distribution with mean zero and (unknown) covariance matrix $\mathbf{\Sigma}$. For a sufficiently large $C > 0$, let S be an ϵ-corrupted set of $n \geq Cd/\epsilon^2$ samples from D in the strong contamination model (Definition 1.1). There exists an algorithm that takes as inputs S and ϵ, runs in $\widetilde{O}(nd/\epsilon^2)$ time and with probability at least 0.99 outputs a unit vector u such that $u^\top \mathbf{\Sigma} u \geq (1 - O(\epsilon \log(1/\epsilon)))\|\mathbf{\Sigma}\|_{\mathrm{op}}$.*

We note that the error guarantee of our algorithm is near-

Authors are listed in alphabetical order. [1]University of Wisconsin-Madison [2]University of California, San Diego. Correspondence to: Ankit Pensia <ankitp@cs.wisc.edu>, Thanasis Pittas <pittas@wisc.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

[1]By nearly linear time algorithms, we mean runtime scaling linearly with (up to logarithmic factors) the size of input, nd.

[2]Naive techniques such as naive pruning and random projection are also susceptible to outliers.

optimal up to logarithmic factors (cf. Appendix B.3). The work most closely related to ours is Jambulapati et al. (2020, Theorem 2), and our algorithm improves upon it in three significant ways:

- (Near-optimal error) The output of Theorem 1.2 achieves the near-optimal $(1 - O(\epsilon \log(1/\epsilon)))$ approximation of $\|\mathbf{\Sigma}\|_{\mathrm{op}}$, whereas their output achieves the sub-optimal approximation of $(1 - O(\sqrt{\epsilon \log(1/\epsilon) \log d}))$.[3]
- (Dependence on ϵ in runtime) Our runtime is $\widetilde{O}(nd/\epsilon^2)$, in contrast to their runtime of $\widetilde{O}\left(nd/\epsilon^{4.5} + ndm/\epsilon^{1.5}\right)$, where m is a parameter discussed below that belongs in $[2, d]$. Thus, our runtime improves upon theirs in all cases.
- (Eigenvalue separation) Their algorithm assumes that the m-th largest eigenvalue of $\mathbf{\Sigma}$ is smaller than $\|\mathbf{\Sigma}\|_{\mathrm{op}}$ by $(1 - \widetilde{\Theta}(\epsilon))$ for some $m \in [2, d]$, which appears in their runtime. Thus, they obtain a nearly linear time only if m=polylog(d/ϵ). Theorem 1.2 has no such restriction.

We finally note that, even without corruptions, $\widetilde{\Omega}(d/\epsilon^2)$ samples are necessary for $(1-O(\epsilon \log(1/\epsilon)))$-approximation of $\|\mathbf{\Sigma}\|_{\mathrm{op}}$. Moreover, even if there are no corruptions, the standard Lanczos algorithm achieving such a guarantee runs in time $\widetilde{O}(\frac{nd}{\sqrt{\epsilon}} + \frac{1}{\epsilon})$ (Sachdeva & Vishnoi, 2014, Theorem 10.1). Thus, our result considerably reduces the price of robustness in PCA, and brings it much closer to the clean data setting.

A streaming algorithm The distributed nature of modern data science applications and large-scale datasets often impose the restriction that the complete dataset cannot be stored in the memory. In such scenarios, the streaming model, defined below, is a much more realistic setting:

Definition 1.3 (Single-Pass Streaming Model)**.** *Let S be a fixed set. In the one-pass streaming model, the elements of S are revealed one at a time to the algorithm, and the algorithm is allowed a single pass over these points.*

In the streaming model, the algorithm also needs to optimize the amount of memory that it uses. We still consider corruptions in the data. This means that the input S above is not comprised of i.i.d. samples from a distribution, but comes from a "corrupted" distribution, formalized below:

Definition 1.4 (TV-contamination)**.** *Given a parameter $\epsilon \in (0, 1/2)$ and a distribution class $\mathcal{D}$, the adversary specifies a distribution D' such that there exists $D \in \mathcal{D}$ with $d_{\mathrm{TV}}(D, D') \le \epsilon$. Then the algorithm draws i.i.d. samples from D'. We say that the distribution D' is an ϵ-corrupted version of the distribution D in total variation distance.*

All prior algorithms for robust PCA needed to store the entire dataset in memory, leading to space usage of at least $\widetilde{\Omega}(d^2/\epsilon^2)$.[4] Building on Theorem 1.2, we present the first algorithm for robust PCA that uses $\widetilde{O}_\epsilon(d)$ memory.

Theorem 1.5 (A streaming algorithm for robust PCA; informal version)**.** *Let $\epsilon \in (0, c)$ for a small constant $c > 0$. Let D be a subgaussian distribution with mean zero and (unknown) covariance matrix $\mathbf{\Sigma}$. Let P be an ϵ-corrupted version of D in total variation distance (Definition 1.4). There is a single-pass streaming algorithm that given ϵ, it reads $\mathrm{poly}(d/\epsilon)$ many samples from P, and with probability 0.99, returns a unit vector u such that $u^\top \mathbf{\Sigma} u \ge (1 - O(\epsilon \log(1/\epsilon)))\|\mathbf{\Sigma}\|_{\mathrm{op}}$. Moreover, the algorithm uses memory $(d/\epsilon) \cdot \mathrm{polylog}(d/\epsilon)$ and runs in time $(nd/\epsilon^2)\mathrm{polylog}(d/\epsilon)$.*

Observe that the asymptotic error of the algorithm is near-optimal and the memory usage is nearly linear in d (the size of the output). We refer the reader to Appendix D.3.1 for further discussion on bit complexity.

1.2. Our Techniques

Our approach is to combine the "easy" version of the robust PCA algorithm from Jambulapati et al. (2020); Kong et al. (2020) with the fast mean estimation algorithm of Diakonikolas et al. (2022d) (see also Diakonikolas et al. (2022b)). This "easy" algorithm works essentially by computing the top eigenvector, v, of the empirical covariance matrix. It then projects the samples onto the v-direction and estimates the true covariance in the v-direction. If this is close to the empirical covariance, then this and the fact that errors cannot substantially decrease the empirical covariance in any direction implies that v is close to a principle eigenvector of the true covariance matrix (cf. Lemma 2.7). Otherwise, some small number of corruptions must account for the difference, and these can be algorithmically filtered out. One then repeats this process of filtering out outliers until an approximate principal eigenvector is found.

The issue with this simple algorithm is its runtime. Although each filtering step can be nearly linear time, there is no guarantee that the number of these steps will be small. It is entirely possible that each filtering step removes only a tiny fraction of corruptions with very large projections in the leading eigenvector direction v. To fix this, we use ideas from Diakonikolas et al. (2022d) and make v essentially a random linear combination of the few largest eigenvectors of $\mathbf{\Sigma}_t$ (the empirical covariance at step t), by defining it to be $\mathbf{\Sigma}_t^p w$ instead of the principle eigenvector of the empirical covariance matrix, where w is a random Gaussian vector, and p is a suitably large integer. This prevents an adversary from "hiding" a corruption by making it orthogonal to some particular v. Instead, any corruption substantially contribut-

[3] In particular, $\epsilon \to 0$ as $d \to \infty$ for non-vacuous guarantees.

[4] In experiments, memory usage has been highlighted as a key bottleneck in robust estimation; see Diakonikolas et al. (2017a).

ing to one of the largest eigenvalues of $\mathbf{\Sigma}_t$ is reasonably likely to be filtered out. Our approach is to keep track of the potential function $\mathrm{tr}(\mathbf{\Sigma}_t^{2p+1})$ and show: (i) small value of the potential certifies that a solution vector can be recovered from the empirical covariance, (ii) whenever this certification is not possible, we can reduce the potential by a multiplicative factor of $(1-\Omega(\epsilon))$. This approach presents two main challenges.

The first key challenge, pertinent to the "certification" that was mentioned before, is to obtain optimal error without any restriction on the spectrum of $\mathbf{\Sigma}$ (as in Jambulapati et al. (2020)). A natural stopping condition is when the current $\mathbf{\Sigma}_t$ is comparable (in an appropriate sense) to $\mathbf{\Sigma}$ up to $(1\pm\epsilon)$ factor. Jambulapati et al. (2020) used Schatten-p norm for large enough p to compare $\mathbf{\Sigma}$ and $\mathbf{\Sigma}_t$. However, a simple example (Example 3.1) shows that, even then, any deterministic algorithm relying on $\mathbf{\Sigma}_t$ must take nd^2 time. Our two-pronged solution is to (i) perform a white-box analysis of robust PCA to enforce a stronger stopping condition and (ii) output a *random* leading eigenvector of $\mathbf{\Sigma}_t$ (cf. Section 3.1).

Second, unlike in Diakonikolas et al. (2022d), we cannot quite achieve a runtime of $\widetilde{O}(nd)$. This is essentially because of the choice of p. Assuming that $\|\mathbf{\Sigma}\|_{\mathrm{op}}=1$, naïve filtering can guarantee that our initial $\mathbf{\Sigma}_t$ at $t=0$ has eigenvalues at most $\mathrm{poly}(d)$.[5] Thus, the starting value of our potential is $d^{O(p)}$, and requires $\widetilde{O}(p/\epsilon)$ rounds of filtering (each of which takes $O(pnd)$ time since we need to do p many matrix-vector products with $\mathbf{\Sigma}$), for a total runtime of $\widetilde{O}(ndp^2/\epsilon)$. Now, our algorithm requires p to be at least $\log(d)/\epsilon$ to distinguish between eigenvalues that differ by a $(1+\epsilon)$-factor, as opposed to $p=O(\log(d))$ in Diakonikolas et al. (2022d), resulting in a runtime of $\widetilde{O}(nd/\epsilon^3)$.

This can be improved to $\widetilde{O}(nd/\epsilon^2)$ by noting that such fine differences in the eigenvalues become relevant only at the last stage ("certification stage") of our algorithm. Until then, we can use a smaller value of p. More formally, for any $p'\geq\log d$, we can still decrease the potential multiplicatively until our stopping condition is satisfied. Although this no longer certifies that we have a good solution, we can use it to upper bound the potential by $\mathrm{poly}(d)$. Thus, we run multiple stages: in stage k, we take $p_k=2^k\log d$ and reduce the potential until it becomes d^C, which takes $\tilde{O}(ndp_k/\epsilon)$ time. By the time we reach $p=\log(d)/\epsilon$, since that stage also starts with potential at most d^C, the algorithm terminates in $\tilde{O}(nd/\epsilon^2)$ time, overall.

Finally, as in Diakonikolas et al. (2022d), this algorithm can be turned into a streaming one. Since we use a small number of filters, where each filter removes all samples x with $|v^\top x|>T$ (for some carefully chosen v and T), we can implement this with low memory by storing just the v's and T's of the previously created filters, along with a minimal set of temporary variables for necessary calculations.

[5] As all quantities scale with $\|\mathbf{\Sigma}\|_{\mathrm{op}}$, we use this normalization for the sake of simplifying presentation.

1.3. Related Work

Our work is situated within the field of algorithmic robust statistics, where the goal is to develop computationally-efficient algorithms for high-dimensional estimation problems that are robust to outliers. We refer the reader to a recent book (Diakonikolas & Kane, 2023) on this topic. The most related line of works to our paper is the work on high-dimensional robust PCA, starting from Xu et al. (2013). Subsequently, polynomial-time algorithms with near-optimal dependence on ϵ in the error guarantee were developed in Jambulapati et al. (2020); Kong et al. (2020). We note that these algorithms had runtime at least $\Omega(nd^2)$. Jambulapati et al. (2020) also developed a nearly linear time algorithm for robust PCA; however, their algorithm runs in nearly linear-time only in certain cases, and the dependence on ϵ in the error guarantee is sub-optimal and dimension-dependent. Finally, our streaming algorithm is inspired by Diakonikolas et al. (2022d), who presented the first streaming algorithms for various high-dimensional robust estimation tasks.

We remark that robust PCA is closely related to the problem of robust covariance estimation in the operator norm, where the algorithm is required to output an estimate $\widehat{\mathbf{\Sigma}}$ given a corrupted set of samples such that $\|\mathbf{\Sigma}-\widehat{\mathbf{\Sigma}}\|_{\mathrm{op}}$ is small. It is easy to see that the top eigenvector of $\widehat{\mathbf{\Sigma}}$ satisfies the guarantees of robust PCA. Although the sample complexity of robust covariance estimation in operator norm is still $\tilde{O}(d/\epsilon^2)$ for Gaussian data, Diakonikolas et al. (2017b, Theorem 1.5) has shown that the problem is computationally hard (in the statistical query model) unless one takes $\Omega(d^2)$ samples. With $\Omega(d^2)$ samples, one can use robust mean estimation algorithm to estimate the second moment of the samples in Frobenius norm (and thus in operator norm); see the discussion in Diakonikolas et al. (2017b). Thus, robust PCA is an easier task, both computationally and statistically, than robust covariance estimation in operator norm, while still being useful.

We discuss additional related work in Appendix A.

2. Preliminaries

Notation We denote $[n]:=\{1,\dots,n\}$. For a vector v, we let $\|v\|_2$ and $\|v\|_\infty$ denote its ℓ_2 and infinity-norm respectively. We use boldface capital letters for matrices. We use $\mathbf{I}$ for the identity matrix. For two matrices $\mathbf{A}$ and $\mathbf{B}$, we use $\mathrm{tr}(\mathbf{A})$ for the trace of $\mathbf{A}$, and $\langle\mathbf{A},\mathbf{B}\rangle:=\mathrm{tr}(\mathbf{A}^\top\mathbf{B})$ for the trace-inner product. We use $\|\mathbf{A}\|_{\mathrm{F}}, \|\mathbf{A}\|_{\mathrm{op}}, \|\mathbf{A}\|_p$ for the Frobenius, operator, and Schatten p-norm of a matrix $\mathbf{A}$ for $p\geq 1$. In particular, if $\mathbf{A}$ is a $d\times d$ symmetric matrix

with eigenvalues $\lambda_1, \ldots, \lambda_d$, then $\|\mathbf{A}\|_p := (\sum_i |\lambda_i|^p)^{1/p}$. We say that a square symmetric matrix $\mathbf{A}$ is PSD (positive semidefinite), and write $\mathbf{A} \succeq 0$, if $x^\top \mathbf{A} x \geq 0$ for all $x \in \mathbb{R}^d$. We write $\mathbf{A} \preceq \mathbf{B}$ when $\mathbf{B} - \mathbf{A}$ is PSD. We write $a \lesssim b$, when there exists an absolute universal constant $C > 0$ such that $a \leq Cb$.

For a distribution D over a domain $\mathcal{X}$ and a weight function $w : \mathcal{X} \to \mathbb{R}_+$, we use D_w to denote the distribution over $\mathbb{R}^d$ with pdf $D_w(x) := D(x)w(x)/\int D(x)w(x)$. We denote the second moment of D by $\mathbf{\Sigma}_D := \mathbf{E}_{X\sim D}[XX^\top]$. We will repeatedly use that $\mathbf{E}_{x\sim D}[\|\mathbf{U}x\|_2^2] = \langle \mathbf{U}^\top \mathbf{U}, \mathbf{\Sigma}_D \rangle$.

We now collect some facts that we shall use in the paper. Starting with Schatten norms, for a symmetric matrix $\mathbf{A}$, we have $\|\mathbf{A}\|_2 = \|\mathbf{A}\|_{\mathrm{F}}$, $\|\mathbf{A}\|_\infty = \|\mathbf{A}\|_{\mathrm{op}}$, and for a PSD matrix $\mathbf{A}$, $\|\mathbf{A}\|_1 = \mathrm{tr}(\mathbf{A})$. Schatten norms satisfy Hölder inequality for trace inner product: $\langle \mathbf{A}, \mathbf{B} \rangle \leq \|\mathbf{A}\|_p \|\mathbf{B}\|_q$ if $1/p + 1/q = 1$. Moreover, Schatten-p norms decrease with p and are close to each other if p is large.

Fact 2.1. *If $\mathbf{A} \in \mathbb{R}^{d\times d}$ is symmetric and $p \geq 1$, the Schatten norms of $\mathbf{A}$ satisfy $\|\mathbf{A}\|_{p+1} \leq \|\mathbf{A}\|_p \leq \|\mathbf{A}\|_{p+1} d^{\frac{1}{p(p+1)}}$.*

The following concerns the distribution of quadratic polynomial of Gaussians, i.e., $z^\top \mathbf{A} z$ for $z \sim \mathcal{N}(0, \mathbf{I})$.

Fact 2.2. *For any symetric $d \times d$ matrix $\mathbf{A}$, we have $\mathbf{Var}_{z\sim\mathcal{N}(0,\mathbf{I})}[z^\top \mathbf{A} z] = 2\|\mathbf{A}\|_{\mathrm{F}}^2$. If $\mathbf{A}$ is a PSD matrix, then for any $\beta > 0$, $\mathbf{Pr}_{z\sim\mathcal{N}(0,\mathbf{I})}[z^\top \mathbf{A} z \geq \beta \mathrm{tr}(\mathbf{A})] \geq 1 - \sqrt{e\beta}$.*

Finally, we record the guarantee of power iteration:

Fact 2.3 (Power iteration)**.** *For any PSD matrix $\mathbf{A} \in \mathbb{R}^{d\times d}$ and $\epsilon, \delta \in (0, 1)$, if $p > \frac{C}{\epsilon}\log(d/(\epsilon\delta))$ for a sufficiently large constant C, and $u := \mathbf{A}^p z$ for $z \sim \mathcal{N}(0, \mathbf{I})$, then $\mathbf{Pr}\left[u^\top \mathbf{A} u/\|u\|_2^2 \geq (1-\epsilon)\|\mathbf{A}\|_{\mathrm{op}}\right] \geq 1 - \delta$.*

2.1. Stability Condition

Recall that in our notation we use $\mathbf{\Sigma}_D = \mathbf{E}_{X\sim D}[XX^\top]$ for the second moment of a distribution D, and $D_w(x) := D(x)w(x)/\int D(x)w(x)$ for the distribution D re-weighted by $w(x)$. Our algorithm will rely on the following condition.

Definition 2.4 (Stability Condition)**.** *Let $0 < \epsilon < 1/2$ and $\epsilon \leq \gamma < 1$. A distribution G on $\mathbb{R}^d$ is called (ϵ, γ)-stable with respect to a PSD matrix $\mathbf{\Sigma} \in \mathbb{R}^{d\times d}$, if for every weight function $w : \mathbb{R}^d \to [0, 1]$ with $\mathbf{E}_{X\sim G}[w(X)] \geq 1 - \epsilon$, the weighted second moment matrix, $\mathbf{\Sigma}_{G_w} := \mathbf{E}_{X\sim G}[w(x)XX^\top]/\mathbf{E}_{X\sim G}[w(X)]$, satisfies that $(1-\gamma)\mathbf{\Sigma} \preceq \mathbf{\Sigma}_{G_w} \preceq (1+\gamma)\mathbf{\Sigma}$.*

In particular, the second moment matrix is *stable* under deletion of ϵ-fraction. Some remarks are in order: (i) The definition above is intended for distributions with zero mean; This can be assumed without loss of generality since we can always work with pairwise differences, (ii) The uniform distribution over a set of $n = Cd/(\epsilon^2 \log(1/\epsilon))$ i.i.d. samples from a subgaussian distribution (cf. Definition B.1) with covariance $\mathbf{\Sigma}$ is w.h.p. (ϵ, γ)-stable with $\gamma = O(\epsilon \log(1/\epsilon))$ (Jambulapati et al., 2020), and (iii) As stated in Theorem 1.2, the algorithm takes as input an ϵ-corrupted set of samples from a subgaussian distribution. However, for notational convenience, we will consider the input to be a distribution $P = (1-\epsilon)G + \epsilon B$ where G is an (unknown) stable distribution and B is arbitrary. P is meant to be the uniform distribution over the ϵ-corrupted set of samples, G will be the part from the remaining inliers, and B that of outliers.[6] We emphasize that the identity of outliers is unknown to the algorithm.

Our algorithm will remove points iteratively, so we will use binary weights $w(x) \in \{0, 1\}$ to distinguish between the points that we keep $(w(x) = 1)$ and the ones that have been removed $(w(x) = 0)$. Throughout the run of our algorithm, we will ensure that the weights satisfy the following setting:

Setting 2.5. Let $0 < 20\epsilon \leq \gamma < \gamma_0$ for a small constant γ_0. Let P be a mixture distribution $P = (1-\epsilon)G + \epsilon B$ on $\mathbb{R}^d$, where G is $(20\epsilon, \gamma)$-stable distribution with respect to a PSD $d \times d$ matrix $\mathbf{\Sigma}$, and $\mathbf{B}$ is arbitrary. Let $w : \mathbb{R}^d \to \{0, 1\}$ be a weight function with $\mathbf{E}_{X\sim G}[w(X)] \geq 1 - 3\epsilon$.

As we will use projections of points to filter outliers, we need the following implication of stability, concerning the average value of these projections (and thresholded projections), as well as their quantiles and their robust estimates.

Lemma 2.6. *In Setting 2.5, define the following: (i) $g(x) := \|\mathbf{U}x\|_2^2$ for some $\mathbf{U} \in \mathbb{R}^{m\times d}$, (ii) L be the top 3ϵ-quantile of $g(x)$ under P_w (P re-weighted by w) and $S := \{x : x > L\}$, (iii) $\widehat{\sigma} := \mathbf{E}_{X\sim P}[w(X)g(X)\mathbb{1}(g(X) \leq L)]$. Then,*

1. $\left| \mathop{\mathbf{E}}_{X\sim G}[w(X)g(X)] - \left\langle \mathbf{U}^\top \mathbf{U}, \mathbf{\Sigma} \right\rangle \right| \leq 2\gamma \left\langle \mathbf{U}^\top \mathbf{U}, \mathbf{\Sigma} \right\rangle$,
2. $\mathbf{E}_{X\sim G}[w(X)g(X)\mathbb{1}\{X \in S\}] \leq 2.35\gamma \left\langle \mathbf{U}^\top \mathbf{U}, \mathbf{\Sigma} \right\rangle$,
3. $L \leq 1.65(\gamma/\epsilon)\langle \mathbf{U}^\top \mathbf{U}, \mathbf{\Sigma} \rangle$,
4. $\left| \widehat{\sigma} - \left\langle \mathbf{U}^\top \mathbf{U}, \mathbf{\Sigma} \right\rangle \right| \leq 4\gamma \left\langle \mathbf{U}^\top \mathbf{U}, \mathbf{\Sigma} \right\rangle$,
5. $\mathbf{Pr}_{X\sim G}\left[\|X\|_2^2 > 2(d/\epsilon)\|\mathbf{\Sigma}\|_{\mathrm{op}}\right] \leq \epsilon$.

As discussed earlier, the goal is to filter out points until a top eigenvector u of the data set's empirical second moment $\mathbf{\Sigma}_{P_w}$ approximately maximizes the true variance. As in previous work, an easy way to detect when this happens is by comparing the empirical variance along u to the true variance along the same direction (which although unknown, it can be easily robustly estimated using Item 4 above). The following is implicit in Xu et al. (2013); Jambulapati et al. (2020); Kong et al. (2020):

[6] This claim mentioned in (iii) technically needs a proof (see, e.g., Lemma 2.12 in Diakonikolas et al. (2022d)).

Lemma 2.7 (Basic Certificate Lemma). *Consider Setting 2.5. Let u be a unit vector with $u^\top \mathbf{\Sigma}_{P_w} u \geq (1 - O(\gamma))\|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}$. If $u^\top \mathbf{\Sigma} u \geq (1-O(\gamma))u^\top \mathbf{\Sigma}_{P_w} u$, then we have that $u^\top \mathbf{\Sigma} u/\|u\|_2^2 \geq (1 - O(\gamma))\|\mathbf{\Sigma}\|_{\mathrm{op}}$.*

2.2. Filtering

We now recall the standard filtering routine from algorithmic robust statistics literature. The filter uses a score $\tau(x)$ for each point x, indicating how atypical the point is for the distribution. Given a (known) upper bound T for the average scores over only the inliers $\mathbf{E}_{X\sim G}[w(X)\tau(X)]$, if the average score of all (corrupted) points, $\mathbf{E}_{X\sim P}[w(X)\tau(X)]$, is much bigger than T, then points with large scores are likely to be outliers. Thus, Algorithm 1 removes all points with scores greater than a (random) threshold.[7] We also note:

1. Instead of storing a bit $w(x)$ for every x, we can store a more succinct description of the filters, i.e., just the r_ℓ's of line 4, and calculate $w(x)$ whenever needed. This modification is amenable to the streaming setting.
2. Since every time we remove all points with $\tau(x) > r_\ell$ and $r_\ell \sim \mathcal{U}([0, r_{\ell-1}])$, this (in expectation) halves the range of $\tau(x)$, and thus Algorithm 1 terminates after $O(\log(\max_x \tau(x)/\min_x \tau(x)))$ steps in expectation.

The following result states that Algorithm 1 removes more outliers than inliers in expectation.

Lemma 2.8 (Guarantees of Filtering). *Let T be such that $(1-\epsilon)\,\mathbf{E}_{X\sim G}[w(x)\tau(x)] < T$ and $\widehat{T}$ such that $|\widehat{T}-T| < T/5$. Denote by F the randomness of* HARDTHRESHOLDINGFILTER *(i.e., the collection of the random thresholds $r_1, r_2 \ldots$ used). Let w' be the weight function returned. Then, (i) $\mathbf{E}_{X\sim P}[w'(X)\tau(X)] \leq 3T$ almost surely, and (ii) $\mathbf{E}_F[\epsilon\,\mathbf{E}_{X\sim B}[w(X)-w'(X)]] > (1-\epsilon)\,\mathbf{E}_{X\sim G}[w(X)-w'(X)]]$.*

Algorithm 1 HARDTHRESHOLDINGFILTER

1: **Input**: Distribution $P = (1-\epsilon)G + \epsilon B$, weights w, scores τ, parameters $\widehat{T}, R$.
2: $w_0(x) \leftarrow w(x)$, and $r_0 \leftarrow R$, $\ell \leftarrow 1$.
3: **while** $\mathbf{E}_{X\sim P}[w(X)\tau(X)] > \frac{5}{2}\widehat{T}$
4: Draw $r_\ell \sim \mathcal{U}([0, r_{\ell-1}])$.
5: $w_{\ell+1}(x) \leftarrow w_\ell(x) \cdot \mathbb{1}(\tau(x) > r_\ell)$.
6: $\ell \leftarrow \ell + 1$.
7: **return** $w_\ell(x)$.

Our main algorithm will repeatedly call HARDTHRESHOLDINGFILTER with appropriate $\tau(x)$ and other parameters. From a technical standpoint, if the second part of Lemma 2.8 (which states that more mass is removed from outliers than inliers) were true deterministically, we would have that $\mathbf{E}_{X\sim G}[w(X)] \geq 1-\epsilon$ no matter how many times the filter is called. This would mean that Setting 2.5 would be maintained throughout the main algorithm. However, part (ii) of Lemma 2.8 holds only in expectation (with respect to F), but one can still show via a martingale argument that a relaxed condition of $\mathbf{E}_{X\sim G}[w(X)] \geq 1 - 3\epsilon$ will still be true throughout the main algorithm w.h.p. The details of this can be found in the full version of the paper (Lemma B.17). For the purposes of the main body, the reader can simply think that Setting 2.5 will always hold.

[7] Consequently, for each point x, $\mathbb{P}(x \text{ is removed}) \propto \tau(x)$.

3. Robust PCA in Nearly-Linear Time

For simplicity of presentation, we focus on proving a weaker version of Theorem 1.2 with runtime $\widetilde{O}(nd/\gamma^3)$ (where γ is the stability parameter; recall $\gamma = O(\epsilon \log(1/\epsilon))$ for subgaussians). This still improves upon prior work in terms of both runtime and error guarantee, while effectively showing key aspects of our approach. In Section 3.3, we outline how we further improve the runtime to $\widetilde{O}(nd/\gamma^2)$.

As mentioned earlier in Sections 1.2 and 2, we run an iterative algorithm, where in each iteration t, we assign a score to each point x, and use these scores to removes points. These scores are usually projections of points along a direction, where outliers have much larger projections than the inliers. Under the stability condition, we can reliably filter outliers as long as the empirical (corrupted) variance is $(1 + C\gamma)$ times larger than the true variance in a direction.

As each round of filtering decreases the variance, we see that $\mathbf{\Sigma}_t$ should decrease with t (after appropriate normalization). As noted in prior work, using scores that involve projections of the samples on the top eigenvector v_t of the empirical second moment $\mathbf{\Sigma}_t$ does not necessarily make good progress because the filtering removes points only in a single (fixed) direction of the largest variance (cf. Section 1.2 for details). Instead, one needs to filter along many of these large variance directions simultaneously (on average), which we do by filtering along the direction $v_t = \mathbf{M}_t z$ for $z \sim \mathcal{N}(0, \mathbf{I})$ and $\mathbf{M}_t = \mathbf{\Sigma}_t^p$ for a large integer $p = \frac{C\log(d/\gamma)}{\gamma}$.

To track progress of the algorithm, we use the potential function $\phi_t := \mathrm{tr}(\mathbf{\Sigma}_t^{2p+1})$. This potential tracks the contribution from all large eigenvalues (and not just the largest one), and has been used in prior work for robust mean estimation. In our setting, we will show that (I) small enough ϕ_t (e.g., $\phi_t \leq \|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}/\mathrm{poly}(d^p)$) implies that $\mathbf{\Sigma}$ and $\mathbf{\Sigma}_t$ share the leading eigenspace *without any restriction on the spectrum of* $\mathbf{\Sigma}$, which in turn, certifies that a good solution vector can be produced, and (II) if the spectrum of $\mathbf{\Sigma}$ and $\mathbf{\Sigma}_t$ disagree, then we can decrease ϕ_t multiplicatively (on average). In particular, we will show that $\phi_{t+1} \leq (1-\gamma)\phi_t$ on average. Combining this with the fact that a simple pruning can always ensure $\phi_0 \leq \mathrm{poly}(d^p)\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}$, we can obtain $\phi_t < \|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}/\mathrm{poly}(d)$ after just $t = O(p\log(d)/\gamma)$ rounds.

We now explain the items (I) and (II) above: Section 3.1

formalizes the notion of "shared leading eigenspaces", and Section 3.2 outlines the decrease of the potential function.

3.1. Advanced Certificate Lemma

We begin by summarizing the algorithmic framework of Jambulapati et al. (2020). In particular, we highlight the roadblocks in their framework to remove the eigenvalue separation condition, and our proposed modifications that rely on formalizing the notion of "shared leading eigenspaces".

Roughly speaking, their iterative algorithm stops at an iteration t such that $\|\mathbf{\Sigma}_t\|_p^p \le (1+C\gamma)^p\|\mathbf{\Sigma}\|_p^p$, where γ is the stability parameter.[8] Then, their algorithm tries the top-m eigenvectors of $\mathbf{\Sigma}_t$ for some m and returns the (normalized) eigenvector v that attains the best possible $v^\top\mathbf{\Sigma}v$ (which can be estimated up to small error); see Lemma 2.7. In particular, the last step takes $\widetilde{O}_\gamma(ndm)$ time. However, as the following example shows (by taking $r=d/2$ as $m=\Omega(d)$), this approach cannot give a nearly-linear time algorithm.

Example 3.1 (Hard Example for Schatten p-Norm Stopping Condition)**.** Let $\mathbf{\Sigma}$ be a PSD projection matrix of rank r, then $\mathbf{\Sigma}_t=\mathbf{I}$ satisfies the condition $\|\mathbf{\Sigma}_t\|_p^p \le (1+C\gamma)^p\|\mathbf{\Sigma}\|_p^p$ as long as $p \gtrsim \log(d/r)/\gamma$. However, any deterministic algorithm to identify a good direction of $\mathbf{\Sigma}$ from the eigenvectors of $\mathbf{\Sigma}_t$ must take $m=\Omega(d-r)$.

Thus, Jambulapati et al. (2020) impose an additional assumption that the largest eigenvalue and the m-th largest eigenvalue of $\mathbf{\Sigma}$ are separated for some m. Consequently, their result and their version of certificate is inherently restricted to have dependence on m; see Proposition 4 therein.

We make two crucial changes in our algorithm (for technical reasons, we also make a change of variable to use $2p+1$ instead of p). The first change is to strengthen the stopping condition: instead of reducing the Schatten norm, which is equivalent to $\langle\mathbf{\Sigma}_t,\mathbf{\Sigma}_t^{2p}\rangle \le (1+C\gamma)^{2p+1}\langle\mathbf{\Sigma},\mathbf{\Sigma}^{2p}\rangle$, we stop at a stronger condition [9] of $\langle\mathbf{\Sigma}_t,\mathbf{\Sigma}_t^{2p}\rangle \le (1+C\gamma)\langle\mathbf{\Sigma},\mathbf{\Sigma}_t^{2p}\rangle$. That is, we stop whenever the empirical variance in directions of $\mathbf{\Sigma}_t^p$, $\langle\mathbf{\Sigma}_t,\mathbf{\Sigma}_t^{2p}\rangle$, is comparable to the true variance, $\langle\mathbf{\Sigma},\mathbf{\Sigma}_t^{2p}\rangle$, up to $(1+C\gamma)$ factor (cf. Lemma 2.6 with $\mathbf{U}=\mathbf{\Sigma}_t^p$). Observe that this $(1+C\gamma)$ factor is the best possible factor that we can achieve while comparing the robust and empirical variances. We are able to achieve this stronger stopping condition because until this condition is satisfied, our algorithm will remove outliers along $\mathbf{\Sigma}_t^p$, which will decrease the potential by a multiplicative factor (this will the topic of the next subsection).

Still, the hard example from Example 3.1 continues to satisfies this stronger condition: one must take $m=\Theta(d\gamma)$ if $r=d/(1+\gamma)$. Thus, any deterministic algorithm would require $m=\Omega(nd^2\gamma)$ time, which is quadratic in d. Our second main insight is to randomize the output of the algorithm. That is, even though, a deterministic algorithm would need to try $\Omega(\gamma d)$ many top eigenvectors of $\mathbf{\Sigma}_t$ before finding a high-variance direction of $\mathbf{\Sigma}$, a top eigenvector sampled randomly from the spectrum of $\mathbf{\Sigma}_t$ will suffice with a large constant probability. In particular, we take a Gaussian sample and iteratively multiply it by $\mathbf{\Sigma}_t$ in the style of power iteration. Formally, we prove the following:

Lemma 3.2 (Advanced Certificate Lemma)**.** *In the Setting 2.5 let* $p>\frac{C}{\gamma}\log(\frac{d}{\gamma})$, $\mathbf{M} := (\mathbf{E}_{X\sim P}[w(X)XX^\top])^p$, *and assume* $\langle\mathbf{\Sigma},\mathbf{M}^2\rangle \ge (1-250\gamma)\langle\mathbf{\Sigma}_{P_w},\mathbf{M}^2\rangle$. *If* $u:=\mathbf{M}z$ *for* $z\sim\mathcal{N}(0,\mathbf{I})$, *then with probability at least* 0.9 *(over the randomness of* z*) we have the following:* [10]

$$u^\top\mathbf{\Sigma}_{P_w}u/\|u\|_2^2 \ge (1-\gamma)\|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}, \tag{1}$$

$$u^\top\mathbf{\Sigma}u \ge (1-O(\gamma))u^\top\mathbf{\Sigma}_{P_w}u\,. \tag{2}$$

The above result states that whenever $\langle\mathbf{\Sigma},\mathbf{M}^2\rangle \ge (1-250\gamma)\langle\mathbf{\Sigma}_{P_w},\mathbf{M}^2\rangle$ for large p, we can use Algorithm 2 to obtain a vector u that will satisfy the "basic certificate" from Lemma 2.7 and thus be a good solution. Our main algorithm will thus call Algorithm 2 to output a vector.

Algorithm 2 SAMPLETOPEIGENVECTOR

Input: Distribution P, weights w, parameters ϵ,γ,δ.
$y \leftarrow \mathbf{\Sigma}_{P_w}^{p'}g$ for $p'=\frac{C}{\gamma}\log\left(\frac{d}{\gamma\delta}\right)$ and $g\sim\mathcal{N}(0,\mathbf{I})$.
$\hat{r} \leftarrow y^\top\mathbf{\Sigma}_{P_w}y/\|y\|_2^2$. ▶{cf. Fact 2.3}
Let $\mathbf{M}:=(\mathbf{E}_{X\sim P}[w(X)XX^\top])^p$ for $p=C\frac{\log(d/\gamma)}{\gamma}$.
$u \leftarrow \mathbf{M}z$ for $z\sim\mathcal{N}(0,\mathbf{I})$.
Find $\widehat{\sigma}_u$ such that $|\widehat{\sigma}_u - u^\top\mathbf{\Sigma}u| \le 4\gamma u^\top\mathbf{\Sigma}u$.
If $\widehat{\sigma}_u \ge (1-C\gamma)u^\top\mathbf{\Sigma}_{P_w}u$ and $\frac{u^\top\mathbf{\Sigma}_{P_w}u}{\|u\|_2^2} \ge (1-\gamma)\widehat{r}_t$
  return $u/\|u\|_2$. ▶{c.f. Lemma 2.7}

Proof Sketch of Lemma 3.2 We conclude with an overview of the proof of Lemma 3.2. Part (1) follows directly by Fact 2.3 as it is simply the power iteration step; Thus, we focus on establishing (2). Define the random variable $Y := z^\top\mathbf{M}\mathbf{A}\mathbf{M}z = u^\top\mathbf{A}u$ for $\mathbf{A} := \mathbf{\Sigma} - (1-250\gamma)\mathbf{\Sigma}_{P_w}$. Note that the condition $\langle\mathbf{\Sigma},\mathbf{M}^2\rangle \ge (1-250\gamma)\langle\mathbf{\Sigma}_{P_w},\mathbf{M}^2\rangle$ can be rewritten as $\mathbf{E}[Y]\ge 0$. Moreover, (2) is equivalent to saying Y is not too negative with constant probability.

To show that Y is not too negative, we will show that its variance is small. In particular, we claim that it suffices to show $\mathbf{Var}[Y]\lesssim\gamma^2\|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}^2\|\mathbf{M}\|_{\mathrm{F}}^4$. To see why it is sufficient, Chebyshev's inequality implies that with constant probability

$$Y \gtrsim -\gamma\|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}\|\mathbf{M}\|_{\mathrm{F}}^2 \tag{3}$$

[8] While comparing norms under stability, $(1+C\gamma)$ is necessary.
[9] Please see Appendix B.4 for why this is stronger.
[10] For technical reasons, we take $\mathbf{M}$ to be $(\mathbf{E}_{X\sim P}[w(X)XX^\top])^p = (\mathbf{E}_{X\sim P}[w(X)])^p\,\mathbf{\Sigma}_{P_w}^p$ and not $\mathbf{\Sigma}_{P_w}^p$.

With (3) at hand, we use the definition of Y and to get that with high probability,

$$u^\top \mathbf{\Sigma} u \geq (1-250\gamma)u^\top \mathbf{\Sigma}_{P_w} u - O\left(\gamma \|\mathbf{M}\|_{\mathrm{F}}^2 \|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}\right) \geq (1-O(\gamma))u^\top \mathbf{\Sigma}_{P_w} u\,,$$

where the last step used (1) and that $\|\mathbf{M}\|_{\mathrm{F}}^2/\|u\|_2^2 \leq O(1)$ holds with constant high probability (Fact 2.2).

Thus, it remains to show $\mathbf{Var}[Y] \lesssim \gamma^2 \|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}^2 \|\mathbf{M}\|_{\mathrm{F}}^4$. By Fact 2.2: $\mathbf{Var}[Y] = 2\|\mathbf{MAM}\|_{\mathrm{F}}^2$, and thus we need to show:

Claim 3.3 (Informal). $\|\mathbf{MAM}\|_{\mathrm{F}} \lesssim \gamma \|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}} \|\mathbf{M}\|_{\mathrm{F}}^2$.

Proof sketch of Claim 3.3. Recall the definition of $\mathbf{A}$. We can decompose $\mathbf{\Sigma}_{P_w}$ into the contribution from inliers (which is very close to $\mathbf{\Sigma}$ by stability) and the contribution due to outliers $\mathbf{\Sigma}_B := \mathbf{E}_{X\sim B}[w(X)XX^\top]$. Roughly speaking, we have $\mathbf{\Sigma}_{P_w} \approx (1-\epsilon)\mathbf{\Sigma} + \epsilon \mathbf{\Sigma}_B$, which implies that $\mathbf{A} \approx \gamma \mathbf{\Sigma} + \epsilon \mathbf{\Sigma}_B$ as $\epsilon \leq \gamma$.

By triangle inequality, $\|\mathbf{MAM}\|_{\mathrm{F}} \leq \gamma \|\mathbf{M\Sigma M}\|_{\mathrm{F}} + \epsilon \|\mathbf{M\Sigma}_B\mathbf{M}\|_{\mathrm{F}}$. The first term is easy to upper bound using PSD property of $\mathbf{\Sigma}$ and $\mathbf{M}$ as follows: $\|\mathbf{M\Sigma M}\|_{\mathrm{F}} \lesssim \|\mathbf{\Sigma}\|_{\mathrm{op}} \|\mathbf{M}^2\|_{\mathrm{F}} \lesssim \|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}} \|\mathbf{M}\|_{\mathrm{F}}^2$ by Fact 2.1 and stability. The second term is more involved. Using the decomposition of $\mathbf{\Sigma}_{P_w}$ in the condition $\langle \mathbf{\Sigma}, \mathbf{M}^2\rangle \geq (1-O(\gamma))\langle \mathbf{\Sigma}_{P_w}, \mathbf{M}^2\rangle$, we obtain that $\langle \mathbf{\Sigma}_B, \mathbf{M}^2\rangle \lesssim (\gamma/\epsilon)\langle \mathbf{\Sigma}, \mathbf{M}^2\rangle$. Finally, as $\mathbf{\Sigma}_B$ is a PSD matrix, we obtain $\epsilon \|\mathbf{M\Sigma}_B\mathbf{M}\|_{\mathrm{F}} \lesssim \epsilon \,\mathrm{tr}(\mathbf{M\Sigma}_B\mathbf{M}) = \epsilon \langle \mathbf{\Sigma}_B, \mathbf{M}^2\rangle \lesssim \gamma \langle \mathbf{\Sigma}, \mathbf{M}^2\rangle \leq \gamma \|\mathbf{\Sigma}\|_{\mathrm{op}} \|\mathbf{M}\|_{\mathrm{F}}^2$, leading to the result. ■

3.2. Reducing the Potential Function Multiplicatively

We now describe Algorithm 3, the main component in achieving Theorem 1.2. We follow the notation defined in Algorithm 3. Recall that the distribution P given as input is just the uniform distribution over the (corrupted) data set and is assumed to be of the form $P = (1-\epsilon)G + \epsilon B$ where G is ($C\epsilon$,γ)-stable (see the comment below Definition 2.4).

In this section, we will quantify the progress of our algorithm by keeping track of the potential function $\phi_t := \mathrm{tr}(\mathbf{B}_t^{2p+1})$ in each round t (where $\mathbf{B}_t$ is scaled version of $\mathbf{\Sigma}_t$, defined in Line 8). [11] As mentioned earlier, the correctness of Algorithm 3 is summarized by the following arguments: (i) whenever the condition $\langle \mathbf{\Sigma}, \mathbf{M}_t^2\rangle \geq (1-C\gamma)\langle \mathbf{\Sigma}_t, \mathbf{M}_t^2\rangle$ is true we can output a good solution with SAMPLETOPEIGENVECTOR , (ii) if the condition is false, then ϕ_{t+1} decreases multiplicatively $\phi_{t+1} \leq (1-\gamma)\phi_t$ on average, (iii) if $\phi_t \leq \|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}/\mathrm{poly}(d)$ then the condition from (i) is necessarily true.

We have shown (i) in the previous subsection, (iii) is fairly standard and omitted from the main body (see Lemma C.4). It remains to show (ii), which we do in Claim 3.4 below.

[11] In previous sections, we used $\mathrm{tr}(\mathbf{\Sigma}_t^{2p+1})$ for simplicity. Since our formal proof will need ϕ_t to be naturally decreasing with t, we use the un-normalized second moment $\mathbf{B}_t$ for which $\mathbf{B}_{t+1} \preceq \mathbf{B}_t$.

Before proving Claim 3.4, we outline how these claims imply a version of Theorem 1.2 with runtime $\widetilde{O}(nd/\gamma^3)$ (a detailed proof can be found in Appendix): By the pruning [12] of Line 4, $\phi_0 \leq (d/\epsilon)^{O(p)} \|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}$. Thus, there exists $t_{\mathrm{end}} = O(\log^2(d/\epsilon)/\gamma^2)$ such that after t_{end} rounds, $\phi_{t_{\mathrm{end}}} < (1-\gamma)^{t_{\mathrm{end}}}\phi_0 \leq \|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}/\mathrm{poly}(d)$, which would cause SAMPLETOPEIGENVECTOR to output a good solution. Now, each iteration of the loop can be implemented in $\widetilde{O}(ndp)$ time: The calculation of $\mathbf{M}_t z$ involves p multiplications of z with the empirical second moment matrix, each of which can be done in $O(nd)$ time as $\mathbf{\Sigma}_t z = \sum_x x(x^\top z)$. Thus, the total runtime is $\widetilde{O}(t_{\mathrm{end}} \cdot ndp) = \widetilde{O}(nd/\gamma^3)$.

Algorithm 3 Robust PCA in Small Number of Iterations

1: **Input**: P, ϵ, γ.
2: Let $p = \frac{C\log(d/\gamma)}{\gamma}$, $t_{\mathrm{end}} = \frac{C\log^2(d/\epsilon)}{\gamma^2}$ for large enough C.
3: Find estimator $\widehat{\sigma}_{\mathrm{op}} \in (0.8\|\mathbf{\Sigma}\|_{\mathrm{op}}, 2d\|\mathbf{\Sigma}\|_{\mathrm{op}})$. ▶{c.f. Lemma 2.6.4 with $\mathbf{U} = \mathbf{I}$}
4: Initialize $w_{1,1}(x) = \mathbb{1}\left(\|x\|_2^2 \leq 10\widehat{\sigma}_{\mathrm{op}}(d/\epsilon)\right)$.
5: **for** $t = 1, \ldots, t_{\mathrm{end}}$ **do**
6: Call SAMPLETOPEIGENVECTOR $(P, w_t, \epsilon, \gamma, \frac{1}{t_{\mathrm{end}}})$.
7: Let P_t be the distribution of P weighted by w_t: $P(x)w_t(x)/\mathbf{E}_{X\sim P}[w_t(X)]$.
8: Let $\mathbf{B}_t := \mathbf{E}_{X\sim P}[w_t(X)XX^\top]$ and $\mathbf{M}_t := \mathbf{B}_t^p$. ▶{$\mathbf{M}_t$ does not need to be explicitly computed.}
9: Let $g_t(x) := \|\mathbf{M}_t x\|_2^2$.
10: $v_t \leftarrow \mathbf{M}_t z_t$, where $z_t \sim \mathcal{N}(0, \mathbf{I})$.
11: Let $f_t(x) = (v_t^\top x)^2$.
12: Let L_t be the 3ϵ-quantile of $f_t(\cdot)$ under P_t.
13: $L_t \leftarrow \max\{L_t, (0.1/d)\widehat{\sigma}_{\mathrm{op}}\|v_t\|_2^2\}$ [13]
14: Let $\tau_t(x) = f_t(x)\mathbb{1}(f_t(x) > L_t)$
15: Find $\widehat{\sigma}_t$ such that $|\widehat{\sigma}_t - v_t^\top \mathbf{\Sigma} v_t| \leq 4\gamma v_t^\top \mathbf{\Sigma} v_t$. (e.g., $\widehat{\sigma}_t := \mathbf{E}_{X\sim P}[w_t(X)f_t(X)\mathbb{1}(f_t(X) \leq L_t)]$). ▶{c.f. Item 4 of Lemma 2.6 with $\mathbf{U} = v_t^\top$ }
16: $\widehat{T}_t \leftarrow 2.35\gamma\widehat{\sigma}_t$.
17: $w_{k,t+1} \leftarrow \mathrm{HardThresholdingFilter}(P, w_t, \tau_t, \widehat{T}_t, R)$.
18: **end for**
19: **return** FAIL.

Claim 3.4 (Informal). *In Setting 2.5, if $\langle \mathbf{\Sigma}, \mathbf{M}_t^2\rangle \geq (1 - C\gamma)\langle \mathbf{\Sigma}_t, \mathbf{M}_t^2\rangle$, then $\mathbf{E}[\phi_{t+1}] \leq (1-\gamma)\phi_t$ in Algorithm 3.*

Proof Sketch of Claim 3.4. For simplicity, we assume that scores $f_t(x)$ used by the algorithm (line 11) are the same as the scores $g_t(x)$. Observe that computing $g_t(x) = \|\mathbf{M}_t x\|_2^2$ for all n points would be computationally costly. This is why we use the one-dimensional random projections $f_t(x)$, which are unbiased estimates of $g_t(x)$. A complete proof that uses the estimates f_t can be found in Appendix C.2.

[12] Naïve pruning removes only a few inliers; see Lemma 2.6.5

[13] This ensures a lower bound for $\tau(x)$ whenever it is non-zero. See second bullet in Section 2.2 for why this is needed.

Using our definitions, we calculate the decrease in potential at the $(t+1)$-th step:

$$\begin{aligned}\phi_{t+1} &= \mathrm{tr}\left(\mathbf{B}_{t+1}^{2p+1}\right) \le \mathrm{tr}\left(\mathbf{B}_t^p \mathbf{B}_{t+1} \mathbf{B}_t^p\right) = \mathrm{tr}(\mathbf{M}_t \mathbf{B}_{t+1} \mathbf{M}_t) \\ &= \mathop{\mathbf{E}}_{X\sim P}[w_{t+1}(X) g_t(X)] \\ &= (1-\epsilon) \mathop{\mathbf{E}}_{X\sim G}[w_{t+1}(X) g_t(X)] + \epsilon \mathop{\mathbf{E}}_{X\sim B}[w_{t+1}(X) g_t(X)] \\ &\le \mathop{\mathbf{E}}_{X\sim G}[w_t(X) g_t(X)] + \epsilon \mathop{\mathbf{E}}_{X\sim B}[w_{t+1}(X) g_t(X)] \,, \end{aligned} \tag{4}$$

where the first inequality uses that $\mathbf{B}_{t+1} \preceq \mathbf{B}_t$ is decreasing in PSD order (along with Fact B.4) and the last inequality uses $w_{t+1} \le w_t$. We argue that the RHS above is $(1 + O(\gamma))\langle \mathbf{\Sigma}, \mathbf{M}_t^2 \rangle$: The first term in (4), which corresponds to inliers, can be upper-bounded by $(1+2\gamma)\langle \mathbf{\Sigma}, \mathbf{M}_t^2 \rangle$ using stability (Lemma 2.6.1) For the second term in (4), we use $\tau_t(x) \le g_t(x) + L_t \le g_t(x) + 1.65(\gamma/\epsilon)\langle \mathbf{\Sigma}, \mathbf{M}_t^2 \rangle$, where the last inequality uses Lemma 2.6.3. Moreover, filtering ensures that $\epsilon \mathbf{E}_{X\sim B}[w_{t+1}(X)\tau_t(X)] \le 2.35\gamma\langle \mathbf{\Sigma}, \mathbf{M}_t^2 \rangle$, where we use Lemma 2.8 with $T = 2.35\gamma\langle \mathbf{\Sigma}, \mathbf{M}_t^2 \rangle$, with this choice of T justified by Lemma 2.6.2. Combining these bounds, we obtain

$$\phi_{t+1} \le (1 + O(\gamma))\langle \mathbf{\Sigma}, \mathbf{M}_t^2 \rangle \,. \tag{5}$$

Now, using our assumption $\langle \mathbf{\Sigma}, \mathbf{M}_t^2 \rangle < (1-C\gamma)\langle \mathbf{\Sigma}_t, \mathbf{M}_t^2 \rangle$, we complete the proof as follows:

$$\begin{aligned}\phi_{t+1} &< (1+O(\gamma))\langle \mathbf{\Sigma}, \mathbf{M}_t^2 \rangle < (1+O(\gamma))(1-C\gamma)\langle \mathbf{\Sigma}_t, \mathbf{M}_t^2 \rangle \\ &\le \frac{(1+O(\gamma))(1-C\gamma)}{\mathbf{E}_{X\sim P}[w_t(X)]} \langle \mathbf{B}_t, \mathbf{M}_t^2 \rangle \le (1-\gamma)\phi_t \,, \end{aligned}$$

using $C \gg 1$, $\langle \mathbf{B}_t, \mathbf{M}_t^2 \rangle = \phi_t$, and $\mathbf{E}_{X\sim P}[w_t(X)] \ge 1 - O(\epsilon)$, as filtering does not delete more than $O(\epsilon)$-mass from the inliers (see discussion below Lemma 2.8). ■

We note that Claim 3.4 holds for all $p \ge \log d$; however, Lemma 3.2 requires large p.

3.3. Improving the Dependence on ϵ in Runtime

In the simpler version of the algorithm, the main reason for a large runtime was that (i) the initial potential was $\mathrm{poly}((d/\epsilon)^p)\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}$, (ii) the value of p was $\frac{C}{\gamma}\log(\frac{d}{\gamma})$, (iii) the potential decreases by only $(1-\gamma)$ factor, and (iv) the algorithm terminates when $\phi_t \le \|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}/\mathrm{poly}(d)$. Since (ii)-(iv) are most likely needed, we modify the algorithm to ensure that the initial potential is much smaller: $\mathrm{poly}((d/\epsilon)^{\log(d)})\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}$. If we are able to ensure this cheaply, the algorithm would need only $\mathrm{polylog}(d/\epsilon)/\gamma$ rounds, saving one factor of $1/\gamma$ from the runtime.

The idea is that while we do need (ii), it is only necessary when the algorithm is about to produce a final solution.[14]

[14] Lemma 3.2 (Certificate lemma) requires $p > C\log(d/\gamma)/\gamma$.

Suppose we run Algorithm 3 with $p = \log d$, where the initial potential is indeed $\mathrm{poly}((d/\epsilon)^{\log(d)})\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}$. In Section 3.2, we guaranteed a $(1-\gamma)$-reduction in potential until the condition $\langle \mathbf{\Sigma}, \mathbf{M}_t^2 \rangle \ge (1 - C\gamma)\langle \mathbf{\Sigma}_t, \mathbf{M}_t^2 \rangle$ gets activated. However, even if this condition gets activated, we do not know if Algorithm 2 will succeed as $p \ll \log(d/\gamma)/\gamma$ (c.f. Lemma 3.2). However, when this condition is violated, we see that the final potential has become much smaller: $\phi_t = \langle \mathbf{B}_t, \mathbf{M}_t^2 \rangle \le \langle \mathbf{\Sigma}_t, \mathbf{M}_t^2 \rangle \le (1 + C\gamma)\langle \mathbf{\Sigma}, \mathbf{M}_t^2 \rangle \le (1 + C\gamma)\|\mathbf{\Sigma}\|_{\mathrm{op}}\|\mathbf{B}_t\|_{2p}^{2p}$, where the second step uses that the condition is violated. Using Fact 2.1 to relate $\|\mathbf{B}_t\|_{2p}^{2p}$ and $\phi_t = \|\mathbf{B}_t\|_{2p+1}^{2p+1}$, we obtain $\phi_t \le \mathrm{poly}(d/\epsilon)\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}$.

So far, we have shown that starting with $\phi_0 \le \mathrm{poly}((d/\epsilon)^{\log(d)})\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}$, we can reduce ϕ_t to $\phi_t \le \mathrm{poly}(d)\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}$ after $t \asymp \log^2(d/\epsilon)/\gamma$ rounds. We can then restart the counter t and double $p' = 2p$. The new potential ϕ_0' will be upper bounded using Fact 2.1 as follows:

$$\phi_0' = \|\mathbf{B}_t\|_{2p'+1}^{2p'+1} \le \|\mathbf{B}_t\|_{2p+1}^{2p'+1} = (\phi_t)^{\frac{2p'+1}{2p+1}} = (\phi_t)^{\frac{4p+1}{2p+1}},$$

i.e., it is still $\mathrm{poly}(d)\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p'+1}$. We run the same procedure repetitively until p reaches its final value of $\Theta(\log^2(d/\gamma)/\gamma)$, where the analysis of the previous two sections are applicable, returning a good vector. This leads to a nested-loop algorithm realizing Theorem 1.2.

4. A Streaming Algorithm for Robust PCA

In this section, we consider the problem of robust PCA in the single-pass streaming model. Recall that in this model the algorithm observes the data points one by one, and the algorithm needs to optimize the memory usage.

Drawing inspiration from techniques in Diakonikolas et al. (2022d), we see that the algorithm from the previous section is partly already amenable to the streaming setting: The filters used are of the form $\mathbb{1}(v^\top x > L)$ which have a compact representation of $O(d)$ space (it suffices to store the vector v and the threshold L). The potential-based analysis also showed that we create at most $O(\mathrm{polylog}(d/\epsilon)/\gamma)$ many such filters. The remaining issue is that there is no fixed dataset to iterate over, e.g., one cannot compute $v = \mathbf{M}z = \mathbf{B}^p z$ as in line 10 of our previous algorithm. In particular, we can not compute $\widehat{\mathbf{B}}^p z$ in the streaming model for an estimate $\widehat{\mathbf{B}} \approx \mathbf{B}$. As in Diakonikolas et al. (2022d), our approach is to use iterative products of sample-based quantities that are sufficiently close to their population-level counterparts. That is, we multiply z sequentially by p empirical estimates of $\mathbf{B}_t$, each calculated over a different set of samples (each multiplication can indeed be implemented in the streaming model). Thus, we approximate $\mathbf{M}$ with $\widehat{\mathbf{M}} := \prod_{\ell=1}^{p} \widehat{\mathbf{B}}_{t,\ell}$, and it suffices as long as $\|\widehat{\mathbf{M}} - \mathbf{M}_t\|_{\mathrm{op}} \le \delta\|\mathbf{M}_t\|_{\mathrm{op}}$ for an appropriate error δ.

The main new technical ingredient in our paper is that we also need $\|\widehat{\mathbf{M}}z\|_2^2/\|z\|_2^2 \geq (1-O(\gamma))\|\mathbf{M}_t\|_{\mathrm{op}}^2$ w.h.p. for $z \sim \mathcal{N}(0,\mathbf{I})$ as opposed to a constant factor approximation in Diakonikolas et al. (2022d). Although a naïve attempt to prove this requires the approximation δ to be too small, leading to sample complexity $\Omega(d^3)$, we are able to improve the dependence on d for this in Lemma D.8. See Appendix D for the complete algorithm and its proof.

5. Conclusion

We gave the first nearly-linear time algorithm for robust PCA that attains near-optimal error guarantees, without eigenvalue gap assumptions in the spectrum of Σ. Additionally, we presented the first sub-quadratic space streaming algorithm for robust PCA.

In terms of future improvements, one potential avenue for optimization is the calculation of $\mathbf{B}^p$ for a PSD matrix $\mathbf{B}$ and $p = \tilde{\Theta}(1/\epsilon)$. It is likely that these matrix-dot products can be approximated faster through the use of Chebyshev approximation (Sachdeva & Vishnoi, 2014, Chapter 10). Additionally, it remains to be determined if the runtime of our algorithm improves in the presence of a gap among the large eigenvalues of the true covariance matrix. Lastly, as highlighted by Diakonikolas et al. (2022d), designing streaming algorithms with optimal sample complexities and robustness to strong contamination is an open problem.

Acknowledgments

Ilias Diakonikolas is supported by NSF Medium Award CCF-2107079, NSF Award CCF-1652862 (CAREER), a Sloan Research Fellowship, and a DARPA Learning with Less Labels (LwLL) grant. Daniel M. Kane is supported by NSF Medium Award CCF-2107547, NSF Award CCF-1553288 (CAREER), and a Sloan Research Fellowship. Ankit Pensia is supported by NSF Award CCF-1652862 (CAREER), and NSF grants CCF-1841190 and CCF-2011255. Thanasis Pittas is supported by NSF Medium Award CCF-2107079.

References

Bakshi, A., Diakonikolas, I., Jia, H., Kane, D. M., Kothari, P. K., and Vempala, S. S. Robustly learning mixtures of k arbitrary gaussians. In *Proc. 54th Annual ACM Symposium on Theory of Computing (STOC)*, 2022.

Balakrishnan, S., Du, S. S., Li, J., and Singh, A. Computationally Efficient Robust Sparse Estimation in High Dimensions. In *Proc. 30th Annual Conference on Learning Theory (COLT)*, volume 65, pp. 1–44, 2017.

Bienstock, D., Jeong, M., Shukla, A., and Yun, S. Robust streaming PCA. In *Advances in Neural Information Processing Systems 35 (NeurIPS)*, 2022.

Candès, E. J., Li, X., Ma, Y., and Wright, J. Robust principal component analysis? *J. ACM*, 58(3), jun 2011. ISSN 0004-5411. doi: 10.1145/1970392.1970395.

Cheng, Y., Diakonikolas, I., Ge, R., and Woodruff, D. P. Faster algorithms for high-dimensional robust covariance estimation. In *Proc. 32nd Annual Conference on Learning Theory (COLT)*, 2019.

Cheng, Y., Diakonikolas, I., Kane, D. M., Ge, R., Gupta, S., and Soltanolkotabi, M. Outlier-robust sparse estimation via non-convex optimization. In *Advances in Neural Information Processing Systems 35 (NeurIPS)*, 2022.

Cherapanamjeri, Y., Aras, E., Tripuraneni, N., Jordan, M. I., Flammarion, N., and Bartlett, P. L. Optimal Robust Linear Regression in Nearly Linear Time. abs/2007.08137, 2020.

Cherapanamjeri, Y., Tripuraneni, N., Bartlett, P. L., and Jordan, M. I. Optimal Mean Estimation without a Variance. In *Proc. 35th Annual Conference on Learning Theory (COLT)*, 2022.

Depersin, J. and Lecué, G. Robust subgaussian estimation of a mean vector in nearly linear time. *The Annals of Statistics*, 50(1):511–536, 2022.

Diakonikolas, I. and Kane, D. M. *Algorithmic High-Dimensional Robust Statistics*. Cambridge University Press, 2023.

Diakonikolas, I., Kamath, G., Kane, D. M., Li, J., Moitra, A., and Stewart, A. Robust estimators in high dimensions without the computational intractability. In *FOCS*, pp. 655–664, 2016.

Diakonikolas, I., Kamath, G., Kane, D. M., Li, J., Moitra, A., and Stewart, A. Being robust (in high dimensions) can be practical. In *Proc. 34th International Conference on Machine Learning (ICML)*, 2017a.

Diakonikolas, I., Kane, D. M., and Stewart, A. Statistical query lower bounds for robust estimation of high-dimensional gaussians and gaussian mixtures. In *Proc. 58th IEEE Symposium on Foundations of Computer Science (FOCS)*, pp. 73–84, 2017b. doi: 10.1109/FOCS.2017.16.

Diakonikolas, I., Kane, D. M., and Stewart, A. List-decodable robust mean estimation and learning mixtures of spherical gaussians. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018*, pp. 1047–1060, 2018. Full version available at https://arxiv.org/abs/1711.07211.

Diakonikolas, I., Kamath, G., Kane, D. M., Li, J., Steinhardt, J., and Stewart, A. Sever: A Robust Meta-Algorithm for Stochastic Optimization. In *Proc. 36th International Conference on Machine Learning (ICML)*, 2019a.

Diakonikolas, I., Kane, D. M., Karmalkar, S., Price, E., and Stewart, A. Outlier-robust high-dimensional sparse estimation via iterative filtering. In *Advances in Neural Information Processing Systems 32 (NeurIPS)*, 2019b.

Diakonikolas, I., Kong, W., and Stewart, A. Efficient Algorithms and Lower Bounds for Robust Linear Regression. In *Proc. 30th Annual Symposium on Discrete Algorithms (SODA)*, pp. 2745–2754. SIAM, 2019c. doi: 10.1137/1.9781611975482.170.

Diakonikolas, I., Kane, D. M., and Pensia, A. Outlier Robust Mean Estimation with Subgaussian Rates via Stability. In *Advances in Neural Information Processing Systems 33 (NeurIPS)*, 2020.

Diakonikolas, I., Kane, D. M., Karmalkar, S., Pensia, A., and Pittas, T. Robust sparse mean estimation via sum of squares. In *Proc. 35th Annual Conference on Learning Theory (COLT)*, 2022a.

Diakonikolas, I., Kane, D. M., Kongsgaard, D., Li, J., and Tian, K. Clustering mixture models in almost-linear time via list-decodable mean estimation. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 1262–1275, 2022b.

Diakonikolas, I., Kane, D. M., Lee, J. C. H., and Pensia, A. Outlier-Robust Sparse Mean Estimation for Heavy-Tailed Distributions. In *Advances in Neural Information Processing Systems 35 (NeurIPS)*, 2022c.

Diakonikolas, I., Kane, D. M., Pensia, A., and Pittas, T. Streaming algorithms for high-dimensional robust statistics. In *International Conference on Machine Learning*, pp. 5061–5117. PMLR, 2022d. arxiv preprint at https://arxiv.org/abs/2204.12399.

Dong, Y., Hopkins, S. B., and Li, J. Quantum Entropy Scoring for Fast Robust Mean Estimation and Improved Outlier Detection. In *Advances in Neural Information Processing Systems 32 (NeurIPS)*, 2019.

Hopkins, S. B. and Li, J. Mixture models, robustness, and sum of squares proofs. In *Proc. 50th Annual ACM Symposium on Theory of Computing (STOC)*, 2018.

Hopkins, S. B., Li, J., and Zhang, F. Robust and Heavy-Tailed Mean Estimation Made Simple, via Regret Minimization. In *Advances in Neural Information Processing Systems 33 (NeurIPS)*, 2020.

Hu, L. and Reingold, O. Robust mean estimation on highly incomplete data with arbitrary outliers. In *Proc. 24th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2021.

Jambulapati, A., Li, J., and Tian, K. Robust sub-gaussian principal component analysis and width-independent schatten packing. *Advances in Neural Information Processing Systems 33 (NeurIPS)*, 2020. arxiv preprint at https://arxiv.org/abs/2006.06980.

Karmalkar, S., Klivans, A., and Kothari, P. K. List-decodable Linear Regression. In *Advances in Neural Information Processing Systems 32 (NeurIPS)*, 2019.

Klivans, A., Kothari, P. K., and Meka, R. Efficient Algorithms for Outlier-Robust Regression. In *Proc. 31st Annual Conference on Learning Theory (COLT)*, volume 75, pp. 1420–1430. PMLR, 2018.

Kong, W., Somani, R., Kakade, S., and Oh, S. Robust meta-learning for mixed linear regression with small batches. In *Advances in Neural Information Processing Systems 33 (NeurIPS)*, 2020.

Kothari, P. K., Steinhardt, J., and Steurer, D. Robust moment estimation and improved clustering via sum of squares. In *Proc. 50th Annual ACM Symposium on Theory of Computing (STOC)*, pp. 1035–1046. ACM Press, 2018. doi: 10.1145/3188745.3188970.

Lai, K. A., Rao, A. B., and Vempala, S. Agnostic Estimation of Mean and Covariance. In *Proc. 57th IEEE Symposium on Foundations of Computer Science (FOCS)*, pp. 665–674, 2016. doi: 10.1109/FOCS.2016.76.

Liu, A. and Moitra, A. Settling the robust learnability of mixtures of gaussians. In *Proc. 53rd Annual ACM Symposium on Theory of Computing (STOC)*, 2021. doi: 10.1145/3406325.3451084.

Marinov, T. V., Mianjy, P., and Arora, R. Streaming principal component analysis in noisy setting. In *Proc. 35th International Conference on Machine Learning (ICML)*, 2018.

Pensia, A., Jog, V., and Loh, P. Robust regression with covariate filtering: Heavy tails and adversarial contamination. *CoRR*, abs/2009.12976, September 2020.

Prasad, A., Suggala, A. S., Balakrishnan, S., and Ravikumar, P. Robust estimation via robust gradient estimation. *Journal of the Royal Statistical Society Series B*, 82(3): 601–627, July 2020. ISSN 13697412. doi: 10.1111/rssb.12364.

Sachdeva, S. and Vishnoi, N. K. Faster algorithms via approximation theory. *Foundations and Trends® in Theoretical Computer Science*, 9:125–210, 2014.

Vershynin, R. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010.

Vershynin, R. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge University Press, 2018.

Xu, H., Caramanis, C., and Mannor, S. Outlier-robust pca: The high-dimensional case. *IEEE Transactions on Information Theory*, 59(1):546–572, 2013.

Zhu, B., Jiao, J., and Steinhardt, J. Robust estimation via generalized quasi-gradients. *Information and Inference: A Journal of the IMA*, 11(2):581–636, 2021. doi: 10.1093/imaiai/iaab018.

Organization The Appendix is organized as follows: Appendix A mentions other related work. Appendix B includes the proofs and additional standard results that were omitted from the main paper. Appendix C provides the complete algorithm, and the associated proofs, achieving (a more general version of) Theorem 1.2. Finally, Appendix D focuses on the streaming algorithm achieving Theorem 1.5.

A. Additional Related Work

Since the publication of Diakonikolas et al. (2016); Lai et al. (2016), computationally-efficient robust algorithms have been developed for many tasks, including mean estimation (Kothari et al., 2018; Dong et al., 2019; Depersin & Lecué, 2022; Diakonikolas et al., 2020; Hopkins et al., 2020; Hu & Reingold, 2021; Cherapanamjeri et al., 2022; Zhu et al., 2021), sparse estimation (Balakrishnan et al., 2017; Diakonikolas et al., 2019b; Cheng et al., 2022; Diakonikolas et al., 2022a;c), covariance estimation (Cheng et al., 2019), linear regression (Klivans et al., 2018; Diakonikolas et al., 2019c; Pensia et al., 2020; Cherapanamjeri et al., 2020), clustering and list-decodable learning (Diakonikolas et al., 2018; Hopkins & Li, 2018; Kothari et al., 2018; Karmalkar et al., 2019; Liu & Moitra, 2021; Bakshi et al., 2022), and stochastic optimization (Diakonikolas et al., 2019a; Prasad et al., 2020). We refer the reader to the recent book (Diakonikolas & Kane, 2023) for more details.

Finally, we emphasize that the focus of our work and that of Jambulapati et al. (2020); Kong et al. (2020) is unrelated to other works studied with similar terminology: Candès et al. (2011); Marinov et al. (2018); Bienstock et al. (2022).

B. Preliminaries: Omitted Facts

We use the following notion of subgaussianity that was also used in Jambulapati et al. (2020).

Definition B.1. *For a parameter $r \geq 1$, we say a distribution D on $\mathbb{R}^d$ with mean zero and covariance $\mathbf{\Sigma}$ is r-subgaussian if for all unit vectors v and $t > 0$,*

$$\mathop{\mathbf{E}}_{X \sim D}[\exp(tv^\top X)] \leq \exp\left(t^2 r(v^\top \mathbf{\Sigma} v)/2\right)$$

Observe that this notion of subgaussianity is stronger than Vershynin (2018, Definition 3.4.1) because the sub-gaussian proxy in the direction v scales with $v^\top \mathbf{\Sigma} v$. Jambulapati et al. (2020) have shown that if a distribution D is $O(1)$-subgaussian, then a set of Cd/γ^2 i.i.d. samples from D is (ϵ, γ) stable for $\gamma = O(\epsilon \log(1/\epsilon))$; see Jambulapati et al. (2020, Lemma 11).

B.1. Linear Algebraic Facts

Fact B.2 (Hölder Inequality for Schatten Norms)**.** *Let $\mathbf{A}, \mathbf{B}$ be two matrices with dimensions $m \times n$ Let $p \geq 1$. Define $q = \frac{p}{p-1}$ if $p > 1$ and ∞ otherwise, then $\langle \mathbf{A}, \mathbf{B} \rangle \leq \|\mathbf{A}\|_p \|\mathbf{B}\|_q$. In particular, $\langle \mathbf{A}, \mathbf{B} \rangle \leq \|\mathbf{A}\|_{\mathrm{F}} \|\mathbf{B}\|_{\mathrm{F}}$ and $\langle \mathbf{A}, \mathbf{B} \rangle \leq \|\mathbf{A}\|_{\mathrm{op}} \|\mathbf{B}\|_1$.*

Fact B.3. *If $\mathbf{A}, \mathbf{B}, \mathbf{C}$ are symmetric $d \times d$ matrices with $\mathbf{A} \succeq 0$ and $\mathbf{B} \preceq \mathbf{C}$ then $\mathrm{tr}(\mathbf{AB}) \leq \mathrm{tr}(\mathbf{AC})$.*

By using the previous inequality iteratively, one can show the following more general fact.

Fact B.4. *Let $\mathbf{A}, \mathbf{B}$ be PSD matrices with $\mathbf{B} \preceq \mathbf{A}$ and $p \in \mathbb{N}$. Then, $\mathrm{tr}\left(\mathbf{B}^p\right) \leq \mathrm{tr}\left(\mathbf{A}^{p-1}\mathbf{B}\right)$.*

Fact B.5. *If $\mathbf{A} \in \mathbb{R}^{d \times d}$ is a PSD matrix, for any other $\mathbf{B} \in \mathbb{R}^{d \times d}$ it holds $\mathbf{BAB}^\top \succeq 0$.*

Proof. For any vector x, denoting $y := \mathbf{B}^\top x$, we have that $x^\top(\mathbf{BAB}^\top)x = y^\top \mathbf{A} y \geq 0$. ■

Fact B.6. *If $\mathbf{A}$ is a PSD matrix, $\|\mathbf{A}\|_{\mathrm{F}}^2 \leq \mathrm{tr}(\mathbf{A})^2$.*

Proof. Note that $\|\mathbf{A}\|_{\mathrm{F}} = \|\mathbf{A}\|_2$ and $\mathrm{tr}(\mathbf{A}) = \|\mathbf{A}\|_1$. The result then follows by Fact 2.1. ■

Fact B.7. *Let $\mathbf{A}$ be a PSD matrix. Then for any unit vector x and $m \in \mathbb{N}$, $x^\top \mathbf{A}^m x \geq (x^\top \mathbf{A} x)^m$.*

Proof. By spectral decomposition, $\mathbf{A}$ is equal to $\sum_i \lambda_i u_i u_i^\top$ for $\lambda_i \geq 0$, and orthonormal vectors u_i. Furthermore, $\mathbf{A}^m = \sum_i \lambda_i^m u_i u_i^\top$. For any unit vector x, the orthonormality of u_i's imply that $\sum_i (u_i^\top x)^2 = 1$.

We now define the non-negative random variable Z that is equal to λ_i with probability $(u_i^\top x)^2$, which is a valid probability assignment since it is non-negative and sums up to 1. In this definition, we have that $\mathbf{E}[Z] = \sum_i \lambda_i (u_i^\top x)^2 = x^\top \mathbf{A} x$.

Moreover, $\mathbf{E}[Z^m] = \sum_i \lambda_i^m (u_i^\top x)^2 = x^\top (\sum_i \lambda_i^m u_i u_i^\top) x = x^\top \mathbf{A}^m x$. Thus, the desired result is equivalent to saying $\mathbf{E}[Z^m] \geq (\mathbf{E}[Z])^m$ for the non-negative random variable Z, which is satisfied by Jensen's inequality. ■

B.2. Probability Facts

Fact B.8 ((Vershynin, 2010)). *Consider a distribution D on $\mathbb{R}^d$ that is supported in an ℓ_2-ball of radius R from the origin. Let $\mathbf{\Sigma}$ be its second moment matrix and $\mathbf{\Sigma}_N = (1/N)\sum_{i=1}^N X_i X_i^\top$ be the empirical second moment matrix using N i.i.d. samples $X_i \sim D$. There is a constant C such that for any $0 < \epsilon < 1$ and $0 < \tau < 1$, if $N > C\epsilon^{-2}\|\mathbf{\Sigma}\|_2^{-1} R^2 \log(d/\tau)$, we have that $\|\mathbf{\Sigma} - \mathbf{\Sigma}_N\|_{\mathrm{op}} \leq \epsilon \|\mathbf{\Sigma}\|_{\mathrm{op}}$, with probability at least $1 - \tau$.*

B.3. Information-theoretic Error

Let us briefly outline why the best approximate guarantee for subgaussian distributions is of the order $(1 - \tilde{\Theta}(\epsilon))$. We will do so by focusing on Gaussian distributions and establishing a lower bound of $(1 - \Omega(\epsilon))$. Let P be the standard Gaussian distribution in $\mathbb{R}^d$. For a unit vector v and $\epsilon \in (0, 0.1)$, let Q be the distribution $\mathcal{N}(0, \mathbf{I} + \epsilon v v^\top)$. Using simple calculations given below, it can be seen that $d_{\mathrm{TV}}(P, Q) \leq \epsilon/2$: we obtain the following series of inequalities

$$\begin{aligned}
d_{\mathrm{TV}}(Q, P) &\leq \sqrt{d_{\mathrm{KL}}(Q||P)} && \text{(Using Pinsker's inequality)} \\
&= \sqrt{\frac{1}{2}\left(-\log|\mathbf{\Sigma}_Q| + \mathrm{tr}(\mathbf{\Sigma}_Q) - d\right)} && \text{(KL divergence between two Gaussians)} \\
&= \sqrt{\frac{1}{2}\left(-\log(1+\epsilon) + \epsilon\right)} && \text{(Using } \mathbf{\Sigma}_Q = \mathbf{I} + \epsilon v v^\top\text{)} \\
&\leq \epsilon/2. && \text{(since } \log(1+x) \geq x - x^2/2 \text{ for } x \geq 0\text{)}
\end{aligned}$$

Let $\widehat{v}$ be the output of any algorithm. The true variance along the direction $\widehat{v}$ under the distribution Q is $1 + \epsilon(\widehat{v}^\top v)^2$. In particular, the approximation factor obtained by the algorithm outputting $\widehat{v}$ is

$$\frac{\widehat{v}^\top \mathbf{\Sigma}_Q \widehat{v}}{\|\mathbf{\Sigma}_Q\|_{\mathrm{op}}} = \frac{1 + \epsilon(\widehat{v}^\top v)^2}{1 + \epsilon} \leq (1 - 0.5\epsilon)(1 + \epsilon(\widehat{v}^\top v)^2).$$

Since P and Q have total variation distance at most $\epsilon/2$, an adversary, that is allowed to corrupt an ϵ-fraction of samples, can simulate i.i.d. samples of P given i.i.d. samples from Q, and vice versa. Now, suppose the true distribution was Q, but the adversary gives i.i.d. samples from P. Since P contains no information about v whatsoever, no algorithm can identify v from the set of ϵ-corrupted samples. As v could be an arbitrary unit vector, no algorithm can output a $\hat{v}$ such that $|v^\top \widehat{v}| \leq 0.01$ for d larger than a constant. As a result, it is not possible to obtain an approximation better than $(1 - \epsilon)$ for Gaussians because $(1 - 0.5\epsilon)(1 + 0.1\epsilon) \leq (1 - 0.1\epsilon)$.

B.4. Comparison Between the Two Stopping Conditions

Consider the setting in Section 3.1. Let $w : \mathbb{R}^d \to \{0, 1\}$ be weights such that probability of inliers is at least $1 - 3\epsilon$. Let $r = \mathbf{E}_{X \sim P}[w_t(X)]$. Recall that $\mathbf{M} = \left(\mathbf{E}_{X \sim P}[w_t(X) X X^\top]\right) = (\mathbf{E}_{X \sim P}[w_t(X)]\mathbf{\Sigma}_t)^p = r^p \mathbf{\Sigma}_t^p$.

Our algorithm stops when $\langle \mathbf{\Sigma}_t, \mathbf{M}^2 \rangle \leq (1 + C\gamma)\langle \mathbf{\Sigma}, \mathbf{M}^2 \rangle$. We will now show that this stopping condition implies the stopping condition that depends solely on the Schatten norms for large p. Using the definition of the Schatten norm, we begin as follows:

$$\begin{aligned}
\|\mathbf{\Sigma}_t\|_{2p+1}^{2p+1} &= \langle \mathbf{\Sigma}_t, \mathbf{\Sigma}_t^{2p} \rangle = r^{-2p} \langle \mathbf{\Sigma}_t, \mathbf{M}^2 \rangle \\
&\leq (1 + C\gamma) r^{-2p} \langle \mathbf{\Sigma}, \mathbf{M}^2 \rangle && \text{(Using stopping condition)} \\
&\leq (1 + C\gamma) r^{-2p} \|\mathbf{\Sigma}\|_{\mathrm{op}} \|\mathbf{M}\|_{\mathrm{F}}^2 \\
&= (1 + C\gamma) \|\mathbf{\Sigma}\|_{\mathrm{op}} \|\mathbf{\Sigma}_t\|_{2p}^{2p} && \text{(Using definition of } \mathbf{\Sigma}_t\text{)} \\
&\leq (1 + C\gamma) \|\mathbf{\Sigma}\|_{\mathrm{op}} \left(\|\mathbf{\Sigma}_t\|_{2p+1} d^{\frac{1}{2p(2p+1)}}\right)^{2p} && \text{(Using Fact 2.1)} \\
&\leq (1 + 2C\gamma) \|\mathbf{\Sigma}\|_{\mathrm{op}} \|\mathbf{\Sigma}_t\|_{2p+1}^{2p},
\end{aligned}$$

where we use that $p \gg \log(d)/\gamma$ and thus $d^{1/(2p+1)} \le (1+C\gamma)$. Rearranging it, we obtain that

$$\|\mathbf{\Sigma}_t\|_{2p+1} \le (1+2C\gamma)\|\mathbf{\Sigma}\|_{\mathrm{op}},$$

and thus $\|\mathbf{\Sigma}_t\|_{2p+1}^{2p+1} \le (1+2C\gamma)^{2p+1}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}$, which is the corresponding stopping condition that depends solely on the Schatten norms of $\mathbf{\Sigma}_t$ and $\mathbf{\Sigma}$. Thus, the stopping condition of Lemma 3.2 is stronger.

B.5. Omitted Proofs Regarding Stability: Proof of Lemma 2.6

In this section we restate and prove Lemma 2.6. We have not optimized the constants (up to the choice of constants C).

Lemma 2.6. *In Setting 2.5, define the following: (i) $g(x) := \|\mathbf{U}x\|_2^2$ for some $\mathbf{U} \in \mathbb{R}^{m\times d}$, (ii) L be the top 3ϵ-quantile of $g(x)$ under P_w (P re-weighted by w) and $S := \{x \,:\, x > L\}$, (iii) $\widehat{\sigma} := \mathbf{E}_{X\sim P}[w(X)g(X)\mathbb{1}(g(X) \le L)]$. Then,*

1. $\left| \mathop{\mathbf{E}}_{X\sim G}[w(X)g(X)] - \left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle \right| \le 2\gamma \left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle$,
2. $\mathbf{E}_{X\sim G}[w(X)g(X)\mathbb{1}\{X\in S\}] \le 2.35\gamma \left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle$,
3. $L \le 1.65(\gamma/\epsilon)\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\rangle$,
4. $\left|\widehat{\sigma} - \left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle\right| \le 4\gamma \left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle$,
5. $\mathbf{Pr}_{X\sim G}\left[\|X\|_2^2 > 2(d/\epsilon)\|\mathbf{\Sigma}\|_{\mathrm{op}}\right] \le \epsilon$.

We prove each part of the lemma separately. For the first one, we have the following, which holds under the slightly stronger condition $\mathbf{E}_{X\sim G}[w(X)] \ge 1-7\epsilon$ (this is because this version will be needed later on).

Lemma B.9. *Let $0 < 20\epsilon \le \gamma < 1$. Let G be a $(20\epsilon,\gamma)$-stable distribution with respect to $\mathbf{\Sigma}$. Let a matrix $\mathbf{U} \in \mathbb{R}^{m\times d}$ and a function $w : \mathbb{R}^d \to [0,1]$ with $\mathbf{E}_{X\sim G}[w(X)] \ge 1-7\epsilon$. For the function $g(x) = \|\mathbf{U}x\|_2^2$, we have that*

$$(1-1.35\gamma)\left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle \le \mathop{\mathbf{E}}_{X\sim G}[w(X)g(X)] \le (1+\gamma)\left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle .$$

Proof. For the upper bound,

$$\mathop{\mathbf{E}}_{X\sim G}[w(X)g(X)] = \left\langle \mathbf{U}^\top\mathbf{U}, \mathop{\mathbf{E}}_{X\sim G}[w(X)XX^\top]\right\rangle \le (1+\gamma)\mathop{\mathbf{E}}_{X\sim G}[w(X)]\left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle \le (1+\gamma)\left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle,$$

where the first step rewrites it in the notation of trace inner product, the second step uses the stability condition and Fact B.3 with $\mathbf{A} = \mathbf{U}^\top\mathbf{U}$, $\mathbf{B} = \mathbf{E}_{X\sim G}[w(X)XX^\top]$, $\mathbf{C} = \mathbf{E}_{X\sim G}[w(X)](1+\gamma)\mathbf{\Sigma}$, and the last step uses $w(x) \le 1$. The other direction is similar, with the only difference being that we lower bound $\mathbf{E}_{X\sim G}[w(X)] \ge 1-7\epsilon$:

$$\mathop{\mathbf{E}}_{X\sim G}[w(X)g(X)] \ge (1-\gamma)(1-7\epsilon)\left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle \ge (1-\gamma-7\epsilon)\left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle \ge (1-1.35\gamma)\left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle ,$$

where we also used that $\epsilon \le \gamma/20$. ∎

We will need the following intermediate result:

Lemma B.10. *Let $0 < 20\epsilon \le \gamma < 1$. Let G be a $(20\epsilon,\gamma)$-stable distribution with respect to $\mathbf{\Sigma}$. Let a matrix $\mathbf{U} \in \mathbb{R}^{m\times d}$, a function $w : \mathbb{R}^d \to [0,1]$ with $\mathbf{E}_{X\sim G}[w(X)] \ge 1-3\epsilon$, and a set S such that $\mathbf{E}_{X\sim G}[w(X)\mathbb{1}\{X\in S\}] \le 4\epsilon$. Then, for the function $g(x) := \|\mathbf{U}x\|_2^2$, we have the following:*

$$\mathop{\mathbf{E}}_{X\sim G}[w(X)g(X)\mathbb{1}\{X\in S\}] \le 2.35\gamma\left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle .$$

Proof. Let $w'(x) := w(x)\mathbb{1}\{X \notin S\} = w(x) - w(x)\mathbb{1}\{X\in S\}$, and note that $\mathbf{E}_{X\sim G}[w'(X)] = \mathbf{E}_{X\sim G}[w(X)] - \mathbf{E}_{X\sim G}[w(X)\mathbb{1}\{X\in S\}] \ge 1-7\epsilon$ by our assumptions. Thus, we have that

$$\begin{aligned}\mathop{\mathbf{E}}_{X\sim G}[w(X)g(X)\mathbb{1}\{X\in S\}] &= \mathop{\mathbf{E}}_{X\sim G}[w(X)g(X)] - \mathop{\mathbf{E}}_{X\sim G}[w'(X)g(X)] \\ &\le (1+\gamma)\left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle - (1-1.35\gamma)\left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle \\ &= 2.35\gamma\left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle ,\end{aligned}$$

where the second line is the application of Lemma B.9 for $w'(x)$. ∎

Now, Item 2 of Lemma 2.6 follows directly as shown below.

Corollary B.11. *In the setting of Lemma B.10, let a mixture distribution $P = (1-\epsilon)G + \epsilon B$ and $S = \{x : x > L\}$, where L is defined to be the top 3ϵ-quantile of $g(x)$ under P_w (the weighted by w version of P). Then,*

$$\mathop{\mathbf{E}}_{X\sim G}[w(X)g(X)\mathbb{1}\{X \in S\}] \le 2.35\gamma \left\langle \mathbf{U}^\top \mathbf{U}, \mathbf{\Sigma} \right\rangle \ .$$

Proof. In order to apply Lemma B.10, we only have to verify that $\mathbf{E}_{X\sim G}[w(X)\mathbb{1}\{X \in S\}] \le 4\epsilon$. Since $P = (1-\epsilon)G + \epsilon B$ and L is by definition the 3ϵ-quantile of $g(x)$ under P_w, and $\epsilon < 1/10$, we have that

$$\begin{aligned}
\mathop{\mathbf{E}}_{X\sim G}[w(X)\mathbb{1}\{g(X) > L\}] &\le \frac{1}{1-\epsilon} \mathop{\mathbf{E}}_{X\sim P}[w(X)\mathbb{1}\{g(X) > L\}] \\
&\le \frac{1}{1-\epsilon} \mathop{\mathbf{Pr}}_{X\sim P_w}[g(X) > L] \\
&\le \frac{3\epsilon}{1-\epsilon} < 4\epsilon \ . \qquad (6)
\end{aligned}$$

An application of Lemma B.10 completes the proof. ∎

We show Item 3 of Lemma 2.6 below.

Lemma B.12. *Let $0 < 20\epsilon \le \gamma < 1$. Let a mixture distribution $P = (1-\epsilon)G + \epsilon B$ on $\mathbb{R}^d$, where G is $(20\epsilon, \gamma)$-stable distribution with respect to a matrix $\mathbf{\Sigma}$. Let $\mathbf{U} \in \mathbb{R}^{m\times d}$ and a function $w : \mathbb{R}^d \to [0,1]$ with $\mathbf{E}_{X\sim G}[w(X)] \ge 1 - 3\epsilon$. Define $g(x) := \|\mathbf{U}x\|_2^2$ and let L be the top 3ϵ-quantile of $g(x)$ under P_w. Then, $L \le 1.65(\gamma/\epsilon)\langle \mathbf{U}^\top \mathbf{U}, \mathbf{\Sigma}\rangle$.*

Proof. Let $w'(x) := w(x)\mathbb{1}(g(x) > L)$. Note that

$$\begin{aligned}
\mathop{\mathbf{E}}_{X\sim G}[w'(X)] &= \frac{\mathbf{E}_{X\sim P}[w(X)\mathbb{1}(g(X) > L)] - \epsilon\, \mathbf{E}_{X\sim B}[w(X)\mathbb{1}(g(X) > L)]}{1-\epsilon} \\
&\ge \frac{\mathbf{E}_{X\sim P}[w(X)\mathbb{1}(g(X) > L)] - \epsilon}{1-\epsilon} \\
&= \frac{\mathbf{E}_{X\sim P}[w(X)]\, \mathbf{Pr}_{X\sim P_w}[g(X) > L] - \epsilon}{1-\epsilon} \\
&\ge \frac{(1-\epsilon)(1-3\epsilon)3\epsilon - \epsilon}{1-\epsilon} \ge \frac{1.43\epsilon}{1-\epsilon} \ ,
\end{aligned}$$

where the second step uses that $w(x) \le 1$, the fourth step uses that $\mathbf{E}_{X\sim P}[w(X)] \ge (1-\epsilon)\,\mathbf{E}_{X\sim G}[w(X)] \ge (1-\epsilon)(1-3\epsilon)$ and that L is the 3ϵ-quantile of g under P_w, and the last step used that $\epsilon < 1/20$. Using this lower bound on $\mathbf{E}_{X\sim G}[w'(X)]$ with Corollary B.11, we have that

$$\mathop{\mathbf{E}}_{X\sim G_{w'}}[g(X)] = \frac{\mathbf{E}_{X\sim G}[w'(X)g(X)]}{\mathbf{E}_{X\sim G}[w'(X)]} \le \frac{1-\epsilon}{1.43\epsilon} \mathop{\mathbf{E}}_{X\sim G}[w(X)g(X)\mathbb{1}(g(X) > L)] \le \frac{1.65\gamma}{\epsilon} \left\langle \mathbf{U}^\top \mathbf{U}, \mathbf{\Sigma} \right\rangle \ .$$

Since the minimum is less than the average, the result above implies that there exists a point in the support of $G_{w'}$ which is smaller than $1.65(\gamma/\epsilon)\langle \mathbf{U}^\top \mathbf{U}, \mathbf{\Sigma}\rangle$. Since $G_{w'}$ is supported only on points bigger than L, it means that $L \le 1.65(\gamma/\epsilon)\langle \mathbf{U}^\top \mathbf{U}, \mathbf{\Sigma}\rangle$. ∎

We now provide the proof of Item 4 of Lemma 2.6 below.

Lemma B.13 (Variance estimator)**.** *Let $0 < 20\epsilon \le \gamma < 1$. Let a mixture distribution $P = (1-\epsilon)G + \epsilon B$ on $\mathbb{R}^d$, where G is $(20\epsilon, \gamma)$-stable distribution with respect to a matrix $\mathbf{\Sigma}$. Let $\mathbf{U} \in \mathbb{R}^{m\times d}$ and a function $w : \mathbb{R}^d \to [0,1]$ with $\mathbf{E}_{X\sim G}[w(X)] \ge 1 - 3\epsilon$. Define $g(x) := \|\mathbf{U}x\|_2^2$ and let L be the top 3ϵ-quantile of $g(x)$ under P_w. For the estimator $\widehat{\sigma} := \mathbf{E}_{X\sim P}[w(X)g(X)\mathbb{1}(g(X) \le L)]$, we have that*

$$\left|\widehat{\sigma} - \left\langle \mathbf{U}^\top \mathbf{U}, \mathbf{\Sigma} \right\rangle\right| \le 4\gamma \left\langle \mathbf{U}^\top \mathbf{U}, \mathbf{\Sigma} \right\rangle \ .$$

Proof. We consider the contribution to $\widehat{\sigma}$ from the two components of the distribution P separately. For the component G, we first note that $\mathbf{E}_{X\sim G}[w(X)\mathbb{1}(g(X)\le L)] \ge \mathbf{E}_{X\sim G}[w(X)] - \mathbf{E}_{X\sim G}[w(X)\mathbb{1}(g(X) > L)] \ge 1-3\epsilon-4\epsilon = 1-7\epsilon$ (the last step can be shown as in (6)). Thus, by Lemma B.9, we have that

$$\left|\mathop{\mathbf{E}}_{X\sim G}[w(X)g(X)\mathbb{1}(g(X)\le L)] - \left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle\right| \le 1.35\gamma \left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle \ . \tag{7}$$

We now consider the contribution to $\widehat{\sigma}$ from B. We first note that $L \le 1.65(\gamma/\epsilon)\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\rangle$ by Lemma B.12. Therefore,

$$\left|\mathop{\mathbf{E}}_{X\sim B}[w(X)g(X)\mathbb{1}(g(X)\le L)] - \left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle\right| \le L + \left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle \le \left(\frac{1.65\gamma}{\epsilon}+1\right)\left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle \ . \tag{8}$$

Finally, combining Equations (7) and (8) and using that $\epsilon \le \gamma$ yields:

$$\begin{aligned}\left|\mathop{\mathbf{E}}_{X\sim P}[w(X)g(X)\mathbb{1}(g(X)\le L)] - \left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle\right| &\le (1-\epsilon)\left|\mathop{\mathbf{E}}_{X\sim G}[w(X)g(X)\mathbb{1}(g(X)\le L)] - \left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle\right| \\ &\quad + \epsilon\left|\mathop{\mathbf{E}}_{X\sim B}[w(X)g(X)\mathbb{1}(g(X)\le L)] - \left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle\right| \\ &\le 4\gamma\left\langle \mathbf{U}^\top\mathbf{U}, \mathbf{\Sigma}\right\rangle \ .\end{aligned}$$

■

We conclude with the last part of Lemma 2.6.

Claim B.14. *Let $0 < 20\epsilon \le \gamma < 1$. Let G be a $(20\epsilon,\gamma)$-stable distribution with respect to a matrix $\mathbf{\Sigma}$. Then, $\mathbf{Pr}_{X\sim G}\left[\|X\|_2^2 > 2(d/\epsilon)\|\mathbf{\Sigma}\|_{\mathrm{op}}\right] \le \epsilon$.*

Proof. Using Markov's inequality and Lemma B.9 with $\mathbf{U} = \mathbf{I}$

$$\mathop{\mathbf{Pr}}_{X\sim G}\left[\|X\|_2^2 > 2(d/\epsilon)\|\mathbf{\Sigma}\|_{\mathrm{op}}\right] \le \frac{\mathbf{E}_{X\sim G}[\|X\|_2^2]}{2(d/\epsilon)\|\mathbf{\Sigma}\|_{\mathrm{op}}} \le \frac{(1+\gamma)\mathrm{tr}(\mathbf{\Sigma})}{2(d/\epsilon)\|\mathbf{\Sigma}\|_{\mathrm{op}}} \le \epsilon \ .$$

■

We finally restate and prove Lemma 2.7 from Section 2.1.

Lemma B.15. *Let $0 < 20\epsilon \le \gamma < \gamma_0$ for a sufficiently small absolute constant γ_0. Let $P = (1-\epsilon)G + \epsilon B$, where G is a $(20\epsilon,\gamma)$-stable distribution with respect to $\mathbf{\Sigma}$. Let $w : \mathbb{R}^d \to [0,1]$ with $\mathbf{E}_{X\sim G}[w(X)] \ge 1-3\epsilon$, and recall that in our notation $\mathbf{\Sigma}_{P_w} := \mathbf{E}_{X\sim P_w}[XX^\top] = \mathbf{E}_{X\sim P}[w(X)XX^\top]/\mathbf{E}_{X\sim P}[w(X)]$. If u is a vector such that $u^\top\mathbf{\Sigma}_{P_w}u/\|u\|_2^2 \ge (1-\gamma)\|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}$ and $u^\top\mathbf{\Sigma}u \ge (1-O(\gamma))u^\top\mathbf{\Sigma}_{P_w}u$, then $u^\top\mathbf{\Sigma}u/\|u\|_2^2 \ge (1-O(\gamma))\|\mathbf{\Sigma}\|_{\mathrm{op}}$.*

Proof. Since $\mathbf{E}_{X\sim G}[w(X)] \ge 1-3\epsilon$, we can use the stability condition as follows: Let a be the (normalized) top eigenvector of $\mathbf{\Sigma}_{P_w}$ and b be the (normalized) top eigenvector of $\mathbf{\Sigma}$. Using the definition of the operator norm and stability, we first obtain the following lower bound on $\|\mathbf{\Sigma}_{P_w}\|$ in terms of $\|\mathbf{\Sigma}\|_{\mathrm{op}}$:

$$\begin{aligned}\|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}} = a^\top\mathbf{\Sigma}_{P_w}a \ge b^\top\mathbf{\Sigma}_{P_w}b &= b^\top\left(\frac{\mathbf{E}_{X\sim P}[w(X)XX^\top]}{\mathbf{E}_{X\sim P}[w(X)]}\right)b &(9)\\ &\ge (1-\epsilon)b^\top\left(\frac{\mathbf{E}_{X\sim G}[w(X)XX^\top]}{\mathbf{E}_{X\sim P}[w(X)]}\right)b = (1-\epsilon)b^\top\left(\frac{\mathbf{E}_{X\sim G}[w(X)]}{\mathbf{E}_{X\sim P}[w(X)]}\mathbf{\Sigma}_{G_w}\right)b &(10)\\ &\ge (1-\epsilon)(1-3\epsilon)b^\top\mathbf{\Sigma}_{G_w}b \ge (1-4\epsilon)(1-\gamma)b^\top\mathbf{\Sigma}b &(11)\\ &\ge (1-2\gamma)\,b^\top\mathbf{\Sigma}b = (1-2\gamma)\|\mathbf{\Sigma}\|_{\mathrm{op}} \ , &(12)\end{aligned}$$

where the penultimate step used that $\epsilon < \gamma/20$. Combining this with the assumption that $u^\top\mathbf{\Sigma}u$ is large compared to $u^\top\mathbf{\Sigma}_{P_w}u$, we obtain the following:

$$\begin{aligned}\frac{u^\top\mathbf{\Sigma}u}{\|u\|_2^2} &\ge (1-O(\gamma))\frac{u^\top\mathbf{\Sigma}_{P_w}u}{\|u\|_2^2} \ge (1-O(\gamma))(1-\gamma)\|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}} \\ &\ge (1-O(\gamma))(1-\gamma)(1-2\gamma)\|\mathbf{\Sigma}\|_{\mathrm{op}} = (1-O(\gamma))\|\mathbf{\Sigma}\|_{\mathrm{op}} \ .\end{aligned}$$

■

B.6. Filtering

In this subsection, we prove the filter guarantees that were described in Section 2.2. We use a slightly more general version of the filter so that it can be employed later on when we will develop a streaming algorithm. The difference is the introduction of an additional error parameter δ in the filter. This will be useful in Appendix D, where the quantity $\mathbf{E}_{X\sim P}[w(x)\tau(x)]$ as well as $\widehat{T}$ will have to be estimated by averaging samples, thus δ will then account for a small additive error in that approximation. For the non-streaming algorithm, we will use $\delta = 0$

Algorithm 4 HARDTHRESHOLDINGFILTER

Input: Distribution $P = (1-\epsilon)G + \epsilon B$, weights w, scores τ, parameters $\widehat{T}, R, \delta$.
$w_0(x) \leftarrow w(x)$, and $r_0 \leftarrow R$, $\ell \leftarrow 1$.
while $\mathbf{E}_{X\sim P}[w(X)\tau(X)] > \frac{5}{2}(\widehat{T}+\delta)$
  Draw $r_\ell \sim \mathcal{U}([0, r_{\ell-1}])$.
  $w_{\ell+1}(x) \leftarrow w_\ell(x)\cdot \mathbb{1}(\tau(x) > r_\ell)$.
  $\ell \leftarrow \ell + 1$.
return $w_\ell(x)$.

Lemma B.16 (Guarantees of Filtering)**.** *Consider one call of* HARDTHRESHOLDINGFILTER*$(P, w, \tau, \widehat{T}, R, \delta)$. Suppose there exists T such that $(1-\epsilon)\,\mathbf{E}_{X\sim G}[w(x)\tau(x)] < T$ and $\widehat{T}$ such that $|\widehat{T} - T| < T/5 + \delta$. Denote by F the randomness of the filter (i.e., the collection of the random thresholds $r_1, r_2 \ldots$ used). Let w' be the weight function returned. Then,*

1. $\mathbf{E}_{X\sim P}[w'(X)\tau(X)] \le 3T + 5d$ *with probability 1 over the randomness of the filter $\mathcal{F}$.*
2. $\mathbf{E}_F[\epsilon\,\mathbf{E}_{X\sim B}[w(X) - w'(X)]] > (1-\epsilon)\,\mathbf{E}_{X\sim G}[w(X) - w'(X)]]$.

Proof. The first part of the lemma follows by the termination condition $\mathbf{E}_{X\sim P}[w'(X)\tau(X)] \le (5/2)(\widehat{T}+\delta)$ and the assumption $|\widehat{T} - T| < T/5 + \delta$.

We show that the second desideratum holds for every filtering round, conditioned on the past rounds. Consider the ℓ-th round of filtering. The probability that a point gets removed (over the random selection of r_ℓ) is $\tau(x)/r_{\ell-1}$. As a result, for any point x that has not been filtered, the probability that a point that gets removed is equal to $\mathbf{E}[(w_\ell(x) - w_{\ell+1}(x))] = \tau(x)/r_{\ell-1}$ Also, note that since the weights are binary decreasing functions, $w_\ell(x) - w_{\ell+1}(x) = w_\ell(x)(w_\ell(x) - w_{\ell+1}(x))$. For the inliers, we have that

$$\begin{aligned}
\mathop{\mathbf{E}}_{r_\ell}\Big[(1-\epsilon)\mathop{\mathbf{E}}_{X\sim G}[w_\ell(X) - w_{\ell+1}(X)]\Big] &= \mathop{\mathbf{E}}_{r_\ell}\Big[(1-\epsilon)\mathop{\mathbf{E}}_{X\sim G}[w_\ell(X)(w_\ell(X) - w_{\ell+1}(X))]\Big] \\
&= (1-\epsilon)\mathop{\mathbf{E}}_{X\sim G}\Big[w_\ell(X)\mathop{\mathbf{E}}_{r_\ell}[w_\ell(X) - w_{\ell+1}(X)]\Big] \\
&= \frac{1-\epsilon}{r_{\ell-1}}\mathop{\mathbf{E}}_{X\sim G}[w_\ell(X)\tau(X)] < \frac{T}{r_{\ell-1}}\,.
\end{aligned}$$

We now turn to the outliers. If the filtering hasn't stopped, it means that the stopping condition of the while loop is still violated and thus $\mathbf{E}_{X\sim P}\,[w_\ell(X)\tau(X)] > (5/2)(\widehat{T}+\delta) > 2T$, (since we have assumed $|\widehat{T} - T| < T/5 + \delta$). Thus,

$$\begin{aligned}
\mathop{\mathbf{E}}_{r_\ell}\Big[\epsilon\mathop{\mathbf{E}}_{X\sim B}[w_\ell(X) - w_{\ell+1}(X)]\Big] &= \mathop{\mathbf{E}}_{r_\ell}\Big[\epsilon\mathop{\mathbf{E}}_{X\sim B}[w_\ell(X)(w_\ell(X) - w_{\ell+1}(X))]\Big] \\
&= \mathop{\mathbf{E}}_{X\sim P}\Big[w_\ell(X)\mathop{\mathbf{E}}_{r_\ell}[w_\ell(X) - w_{\ell+1}(X)]\Big] \\
&\quad - (1-\epsilon)\mathop{\mathbf{E}}_{X\sim G}\Big[w_\ell(X)\mathop{\mathbf{E}}_{r_\ell}[w_\ell(X) - w_{\ell+1}(X)]\Big] \\
&= \frac{1}{r_{\ell-1}}\mathop{\mathbf{E}}_{X\sim P}[w_\ell(X)\tau(X)] - \frac{1-\epsilon}{r_{\ell-1}}\mathop{\mathbf{E}}_{X\sim G}[w_\ell(X)\tau(X)] \\
&> \frac{2T}{r_{\ell-1}} - \frac{T}{r_{\ell-1}} = \frac{T}{r_{\ell-1}}\,.
\end{aligned}$$

■

We now provide the guarantees of the filtering algorithm over the course of the entire execution of RobustPCA.

Lemma B.17. *Let $0 < 20\epsilon \le \gamma < 1/20$. Let $P = (1-\epsilon)G + \epsilon B$, where G is a $(20\epsilon, \gamma)$-stable distribution with respect to $\mathbf{\Sigma}$. Consider an execution of* RobustPCA. *With probability at least 0.2 over the randomness of the algorithm, we have that $(1-\epsilon)\,\mathbf{E}_{X\sim G}[1 - w_{k,t}(X)] + \epsilon\,\mathbf{E}_{X\sim B}[w_{k,t}(X)] \le 2.9\epsilon$.*

Proof. For notational simplicity, first define an ordering over the multi-indices $\{(k,t) : k \in \{1,\dots,k_{\mathrm{end}}\}, t \in \{1,\dots,t_{\mathrm{end}}\}\}$ in the natural way: $(k',t') \le (k,t)$ is defined to mean that either $k' \le k$ or $k' = k$ and $t' \le t$. Also define a successor operator $\mathrm{s}(\cdot)$ that returns the next multi-index according to the way the two loops are structured in RobustPCA, that is, $\mathrm{s}(k,t) = (k,t+1)$ if $t < t_{\mathrm{end}}$ and $\mathrm{s}(k,t) = (k+1,0)$ otherwise.

Let (k^*,t^*) be the first time that $(1-\epsilon)\,\mathbf{E}_{X\sim G}[1 - w_{k,t}(X)] + \epsilon\,\mathbf{E}_{X\sim B}[w_{k,t}(X)] \ge 2.9\epsilon$ (if there exists one, otherwise set $(k^*,t^*) = (k_{\mathrm{end}}, t_{\mathrm{end}})$). Define the sequence

$$\Delta_{(k,t)} = (1-\epsilon) \mathop{\mathbf{E}}_{X\sim G}\left[1 - w_{\min\{(k,t),(k^*,t^*)\}}(X)\right] + \epsilon \mathop{\mathbf{E}}_{X\sim B}\left[w_{\min\{(k,t),(k^*,t^*)\}}(X)\right] .$$

We note that $\Delta_{(k,t)}$ is a supermartingale with respect to the randomness of the filter: If $(k,t) < (k^*,t^*)$, then $\mathbf{E}[\Delta_{\mathrm{s}(k,t)} \mid \mathcal{F}_{(k,t)}] \le \Delta_{(k,t)}$, where $\mathcal{F}_{(k,t)}$ denotes the randomness of the filters applied so far. This follows by Lemma B.16, as long as the lemma is applicable: We apply that lemma with $T = 2.35\gamma v_{k,t}^\top \mathbf{\Sigma} v_{k,t}$ and $\widehat{T} = 2.35\gamma\widehat{\sigma}_{k,t}$. Its first requirement that $(1-\epsilon)\,\mathbf{E}_{X\sim G}\left[w_{k,t}(x)\tau_{k,t}(x)\right] < T$ is satisfied by Item 2 of Lemma 2.6 (specifically, the special case that uses $\mathbf{U} = v_{k,t}^\top$). For its second requirement $|\widehat{T} - T| < T/5$ we have that $|\widehat{T} - T| = 2.35\gamma|\widehat{\sigma}_{k,t} - v_{k,t}^\top \mathbf{\Sigma} v_{k,t}| \le 2.35\gamma \cdot 4\gamma \cdot v_{k,t}^\top \mathbf{\Sigma} v_{k,t} \le (1/5)\cdot 2.35\gamma v_{k,t}^\top \mathbf{\Sigma} v_{k,t} = T/5$, where the second step uses Item 4 of Lemma 2.6 (with $\mathbf{U} = v_{k,t}^\top$) and the last inequality uses that $\gamma < 1/20$. All these intermediate lemmata that we just refered to are applicable when $(1-\epsilon)\,\mathbf{E}_{X\sim G}[1 - w_{k,t}(X)] + \epsilon\,\mathbf{E}_{X\sim B}[w_{k,t}(X)] \le 2.9\epsilon$.

Otherwise, if $(k,t) \ge (k^*,t^*)$ we still have $\mathbf{E}[\Delta_{\mathrm{s}(k,t)} \mid \mathcal{F}_{(k,t)}] \le \Delta_{(k,t)}$, since by definition the sequence $\Delta_{\mathrm{s}(k,t)}$ remains unchanged.

In the beginning, we have that $\Delta_{(1,1)} \le 2\epsilon$. This is because we start with ϵ-fraction of outliers and the naïve pruning of line 5 of the algorithm cannot remove more than ϵ mass from the distribution G, as shown in Item 5 of Lemma 2.6.

By Doob's inequality for martingales, we have that

$$\mathbf{Pr}\left[\max_{(1,1)\le(k,t)\le(k_{\mathrm{end}},t_{\mathrm{end}})} \Delta_{(k,t)} > 2.5\epsilon\right] \le \frac{\Delta_{(1,1)}}{2.5\epsilon} \le 0.8 \,. \tag{13}$$

This implies that with probability 0.2, it holds $(1-\epsilon)\,\mathbf{E}_{X\sim G}[1 - w_{k,t}(X)] + \epsilon\,\mathbf{E}_{X\sim B}[w_{k,t}(X)] \le 2.5\epsilon$ throughout the algorithm's iterations (which is already a bit stronger than the conclusion of the lemma that wanted to show). To see that, assume the contrary, and define $(\widetilde{k},\widetilde{t})$ to be the first time that $(1-\epsilon)\,\mathbf{E}_{X\sim G}[1 - w_{k,t}(X)] + \epsilon\,\mathbf{E}_{X\sim B}[w_{k,t}(X)] > 2.5\epsilon$. Combining that with the trivial observation that $(\widetilde{k},\widetilde{t}) \le (k^*,t^*)$ yields that $\Delta_{(\widetilde{k},\widetilde{t})} > 2.5\epsilon$, which by (13) cannot happen with probability more than 0.2. ∎

C. Robust PCA in Nearly-Linear Time

In this section, we show Theorem C.1 below. Theorem 1.2 follows from Theorem C.1 by recalling that the uniform distribution over a set of $n \gg d/(\epsilon^2\log(1/\epsilon))$ from a subgaussian distribution is stable with parameter $\gamma = O(\epsilon\log(1/\epsilon))$. In contrast to the main body of the paper, where we analyzed a simplified but slower version of our algorithm, in this section we show directly the runtime of $\widetilde{O}(nd/\gamma^2)$. The proof of the simplified (but slower) algorithm is implicit in the arguments below.

Theorem C.1. *Let γ_0 be a sufficiently small positive constant. Let $d > 2$ be an integer, and $\epsilon,\gamma \in (0,1)$ such that $0 < 20\epsilon < \gamma < \gamma_0$. Let P be the uniform distribution over a set of n points in $\mathbb{R}^d$, that can be decomposed as $P = (1-\epsilon)G + \epsilon B$, where G is a $(20\epsilon,\gamma)$-stable distribution (Definition 2.4) with respect to a PSD matrix $\mathbf{\Sigma} \in \mathbb{R}^{d\times d}$. There exists an algorithm that takes as input P,ϵ,γ, runs for $O(\frac{nd}{\gamma^2}\log^4(d/\epsilon))$ time, and with probability at least 0.99, outputs a vector u such that $u^\top \mathbf{\Sigma} u \ge (1-O(\gamma))\|\mathbf{\Sigma}\|_{\mathrm{op}}$.*

Organization This section is organized as follows: Appendix C.1 contains the omitted proofs regarding the "advanced certificate lemma" of Section 3.1, Appendix C.2 proves that the potential decreases by a $(1-\gamma)$ multiplicative factor in every iteration of the algorithm. Finally, Appendix C.3 puts together the previous arguments to complete the proof of Theorem C.1.

C.1. Advanced Certificate Lemma: Omitted Proofs from Section 3.1

We restate and prove the formal version of certificate lemma (Lemma 3.2) below:

Lemma C.2. *Let a sufficiently large constant C. Let $0 < 20\epsilon \le \gamma < \gamma_0$ for a sufficiently small absolute constant γ_0. Let $P = (1-\epsilon)G + \epsilon B$, where G is a $(20\epsilon, \gamma)$-stable distribution with respect to $\mathbf{\Sigma}$. Let $w : \mathbb{R}^d \to [0,1]$ with $\mathbf{E}_{X\sim G}[w(X)] \ge 1 - 3\epsilon$, $p > C\log(d/\gamma)/\gamma$, $\mathbf{M} := (\mathbf{E}_{X\sim P}[w(X)XX^\top])^p$, and assume $\langle \mathbf{\Sigma}, \mathbf{M}^2\rangle \ge (1-250\gamma)\langle \mathbf{\Sigma}_{P_w}, \mathbf{M}^2\rangle$. If $u := \mathbf{M}z$ for a random $z \sim \mathcal{N}(0, \mathbf{I})$, then with probability at least 0.9 (over the random selection of z) we have that:*

1. $u^\top \mathbf{\Sigma}_{P_w} u / \|u\|_2^2 \ge (1-\gamma)\|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}$,
2. $u^\top \mathbf{\Sigma} u > (1 - O(\gamma)) u^\top \mathbf{\Sigma}_{P_w} u$.

Proof. The part of the conclusion that $u^\top \mathbf{\Sigma}_{P_w} u / \|u\|_2^2 \ge (1-\gamma)\|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}$, follows directly by the guarantee of power iteration (Fact 2.3 with probability of failure being less than 0.001).

We now focus on showing that $u^\top \mathbf{\Sigma} u > (1 - O(\gamma)) u^\top \mathbf{\Sigma}_{P_w} u$. Let $\mathbf{A} := \mathbf{\Sigma} - (1-250\gamma)\mathbf{\Sigma}_{P_w}$ with high probability. We analyze the random variable $Y := z^\top \mathbf{MAM} z$ for $z \sim \mathcal{N}(0, \mathbf{I})$. The assumption $\langle \mathbf{\Sigma}, \mathbf{M}^2\rangle \ge (1-250\gamma)\langle \mathbf{\Sigma}_{P_w}, \mathbf{M}^2\rangle$ means that $\langle \mathbf{M}^2, \Sigma - (1-250\gamma)\mathbf{\Sigma}_{P_w}\rangle = \langle \mathbf{M}^2, \mathbf{A}\rangle \ge 0$. Noting that $\mathbf{E}[Y] = \mathrm{tr}(\mathbf{MAM}) = \langle \mathbf{M}^2, \mathbf{A}\rangle$, we have $\mathbf{E}[Y] \ge 0$. Thus, if variance of Y is not too large, we will have that Y is not too negative with large probability.

We show the following technical result on the variance of Y in Appendix C.1.1.

Claim C.3 (Variance of Y)**.** $\mathbf{Var}(Y) \lesssim \gamma^2 \|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}^2 \|\mathbf{M}\|_{\mathrm{F}}^4$.

Using Claim C.3 along with Chebyshev's inequality, we get that with probability at least 0.999, Y satisfies the following lower bound:

$$Y \ge \mathbf{E}[Y] - \sqrt{1000 \operatorname*{\mathbf{Var}}_{z\sim\mathcal{N}(0,\mathbf{I})}[Y]} = -O\left(\gamma \|\mathbf{M}\|_{\mathrm{F}}^2 \|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}\right) . \tag{14}$$

We need one more intermediate result. For the vector $u := \mathbf{M}z$ for $z \sim \mathcal{N}(0, \mathbf{I})$, an application of Fact 2.2 with $\mathbf{A} = \mathbf{M}^2$ and $\beta = 10^{-4}/e$ yields

$$\mathbf{Pr}\left[\|\mathbf{M}\|_{\mathrm{F}}^2 / \|u\|_2^2 \le 1/\beta\right] = \operatorname*{\mathbf{Pr}}_{z\sim\mathcal{N}(0,\mathbf{I})}\left[z^\top \mathbf{M}^2 z \ge \beta \mathrm{tr}(\mathbf{M}^2)\right] \ge 1 - \sqrt{e\beta} = 0.99 . \tag{15}$$

We can now complete the proof of Lemma C.2. In what follows, we condition on the following events: (i) $\|\mathbf{M}\|_{\mathrm{F}}^2/\|u\|_2^2 \le 1/\beta \le 10^5$, (ii) the inequality of (14) holds, and (iii) the conclusion from part one of the lemma. Since all of these events happen individually with probability 0.99 at least, we have that the intersection of the three events has probability at least 0.97 by a union bound. Recalling that $Y = z^\top \mathbf{MAM} z = u^\top \mathbf{\Sigma} u - (1-250\gamma) u^\top \mathbf{\Sigma}_{P_w} u$ and dividing by $\|u\|_2^2$ both sides of (14), we have the following:

$$\begin{aligned}
\frac{u^\top \mathbf{\Sigma} u}{\|u\|_2^2} &\ge (1-250\gamma)\frac{u^\top \mathbf{\Sigma}_{P_w} u}{\|u\|_2^2} - O\left(\gamma \frac{\|\mathbf{M}\|_{\mathrm{F}}^2}{\|u\|_2^2} \|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}\right) \\
&\ge (1-250\gamma)\frac{u^\top \mathbf{\Sigma}_{P_w} u}{\|u\|_2^2} - O\left(\gamma \|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}\right) && \text{(using that the event from (15) holds)} \\
&\ge (1-250\gamma)\frac{u^\top \mathbf{\Sigma}_{P_w} u}{\|u\|_2^2} - O\left(\frac{\gamma}{1-\gamma}\frac{u^\top \mathbf{\Sigma}_{P_w} u}{\|u\|_2^2}\right) && (u^\top \mathbf{\Sigma}_{P_w} u/\|u\|_2^2 \ge (1-\gamma)\|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}) \\
&= (1 - O(\gamma))\frac{u^\top \mathbf{\Sigma}_{P_w} u}{\|u\|_2^2} . && (\gamma < 1/20)
\end{aligned}$$

Thus, we have shown that the second bullet of Lemma C.2 holds with probability 0.97. The first bullet holds with probability 0.99 when C is appropriately large constant. Thus, by a union bound, both bullets hold with probability at least 0.90. This completes the proof of Lemma C.2. ∎

Finally, we formally show the result that whenever the potential function ϕ is smaller than $\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}/\mathrm{poly}(d/\epsilon)$, then the condition of the previous lemma gets satisfied. Note that the factor $(1-2\gamma)^{2p}$ mentioned in the conclusion of the following lemma is indeed bigger than $1/\mathrm{poly}(d/\epsilon)$ since $p = O(\log(d/\gamma)/\gamma)$.

Lemma C.4. *Let $0 < 20\epsilon \le \gamma < 1/20$, and a positive integer p. Let $P = (1-\epsilon)G + \epsilon B$, where G is a $(20\epsilon, \gamma)$-stable distribution with respect to $\mathbf{\Sigma}$, and let $w : \mathbb{R}^d \to [0,1]$ with $\mathbf{E}_{X\sim G}[w(X)] \ge 1 - 3\epsilon$. Define $\mathbf{B} := \mathbf{E}_{X\sim P}[w(X)XX^\top]$, $\mathbf{M} := \mathbf{B}^p$, and $\phi := \mathrm{tr}(\mathbf{B}^{2p+1})$. If $\phi \le \mathbf{E}_{X\sim P}[w(X)]\frac{(1-2\gamma)^{2p}}{1-250\gamma}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}$, then $\langle \mathbf{\Sigma}, \mathbf{M}^2\rangle \ge (1-250\gamma)\langle \mathbf{\Sigma}_{P_w}, \mathbf{M}^2\rangle$.*

Proof. We claim that for the statement to be true, it suffices to show that

$$\langle \mathbf{M}^2, \mathbf{\Sigma}\rangle \ge (1-2\gamma)^{2p}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1} \,. \tag{16}$$

This is indeed sufficient, because if we had (16) at hand, then whenever $\phi \le \mathbf{E}_{X\sim P}[w(X)]\frac{(1-2\gamma)^{2p}}{1-250\gamma}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}$, we have that

$$\langle \mathbf{M}^2, \mathbf{\Sigma}_{P_w}\rangle = \frac{\phi}{\mathbf{E}_{X\sim P}[w(X)]} \le \frac{(1-2\gamma)^{2p}}{1-250\gamma}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1} \le \frac{1}{1-250\gamma}\langle \mathbf{M}^2, \mathbf{\Sigma}\rangle \,,$$

where the first step uses the definitions $\phi = \mathrm{tr}(\mathbf{B}^{2p+1}) = \langle \mathbf{M}^2, \mathbf{B}\rangle = \mathbf{E}_{X\sim P}[w(X)]\langle \mathbf{M}^2, \mathbf{\Sigma}_{P_w}\rangle$, the second step uses $\phi \le \mathbf{E}_{X\sim P}[w(X)]\frac{(1-2\gamma)^{2p}}{1-250\gamma}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1}$, and the last step uses (16).

In the reminder, we establish (16). Consider the spectral decomposition $\mathbf{\Sigma} = \sum_{i=1}^d \lambda_i u_i u_i^\top$. We have that:

$$\langle \mathbf{M}^2, \mathbf{\Sigma}\rangle = \left\langle \mathbf{M}^2, \sum_{i=1}^d \lambda_i u_i u_i^\top \right\rangle = \sum_{i=1}^d \lambda_i u_i^\top \mathbf{M}^2 u_i \ge \lambda_1 u_1^\top \mathbf{M}^2 u_1 = \|\mathbf{\Sigma}\|_{\mathrm{op}} u_1^\top \mathbf{M}^2 u_1 \ge (1-2\gamma)^{2p}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1} \,, \tag{17}$$

where the last inequality is shown in the reminder of this proof: First, we have that

$$\begin{aligned}\mathbf{\Sigma}_{P_w} &= \frac{\mathbf{E}_{X\sim P}[w(X)XX^\top]}{\mathbf{E}_{X\sim P}[w(X)]} \succeq \frac{(1-\epsilon)\,\mathbf{E}_{X\sim G}[w(X)]}{\mathbf{E}_{X\sim P}[w(X)]}\frac{\mathbf{E}_{X\sim G}[w(X)XX^\top]}{\mathbf{E}_{X\sim G}[w(X)]} \\ &\succeq (1-\epsilon)(1-3\epsilon)(1-\gamma)\mathbf{\Sigma} \succeq (1-2\gamma)\mathbf{\Sigma} \,,\end{aligned} \tag{18}$$

where the first line uses that $P = (1-\epsilon)G + \epsilon B$, and the last line uses $\mathbf{E}_{X\sim G}[w(X)] \ge 1 - 3\epsilon$, $w(x) \le 1$, and stability condition for the second moment of the normalized distribution G, and the last step uses that $\epsilon < \gamma/20$. We now complete the proof of the last step of (17) as follows: Let u_1 be the top eigenvector of $\mathbf{\Sigma}$. We have that

$$u_1^\top \mathbf{\Sigma}_{P_w}^{2p} u_1 \ge (u_1^\top \mathbf{\Sigma}_{P_w} u_1)^{2p} \ge ((1-2\gamma)u_1^\top \mathbf{\Sigma} u_1)^{2p} = (1-2\gamma)^{2p}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p} \,,$$

where the first inequality uses Fact B.7, and the second inequality uses (18).

■

C.1.1. Bound on the Variance of Y

We now prove the following result on the bound of the variance that was omitted from the proof of Lemma C.2 above.

Claim C.3 (Variance of Y)**.** $\mathbf{Var}(Y) \lesssim \gamma^2\|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}^2\|\mathbf{M}\|_{\mathrm{F}}^4$.

Proof. As $Y = z^\top \mathbf{MAM}z$, Fact 2.2 implies that $\mathbf{Var}[Y] = 2\|\mathbf{MAM}\|_{\mathrm{F}}^2$. We will, thus, upper bound $\|\mathbf{MAM}\|_{\mathrm{F}}$ in the remainder of the proof.

Since $P = (1-\epsilon)G + \epsilon B$, we have that $\mathbf{\Sigma}_{P_w} = (1-\epsilon)\mathbf{\Sigma}_{P_w}^G + \epsilon\mathbf{\Sigma}_{P_w}^B$, where $\mathbf{\Sigma}_{P_w}^G := \mathbf{E}_{X\sim G}[w(X)XX^\top]/\,\mathbf{E}_{X\sim P}[w(X)]$, and $\mathbf{\Sigma}_{P_w}^B := \mathbf{E}_{X\sim B}[w(X)XX^\top]/\,\mathbf{E}_{X\sim P}[w(X)]$. Using this decomposition, we obtain the following expression for $\mathbf{A}$:

$$\begin{aligned}\mathbf{A} &= \mathbf{\Sigma} - (1-250\gamma)\left((1-\epsilon)\mathbf{\Sigma}_{P_w}^G + \epsilon\mathbf{\Sigma}_{P_w}^B\right) \\ &= \left(\mathbf{\Sigma} - \mathbf{\Sigma}_{P_w}^G(1-250\gamma-\epsilon+250\gamma\epsilon)\right) - \epsilon(1-250\gamma)\mathbf{\Sigma}_{P_w}^B \,.\end{aligned}$$

In order to simplify notation, let us define $\mathbf{\Delta} := \mathbf{\Sigma} - \mathbf{\Sigma}_{P_w}^G(1-250\gamma-\epsilon+250\gamma\epsilon)$, and $\widetilde{\epsilon} := \epsilon(1-250\gamma)$. We thus have $\mathbf{A} = \mathbf{\Delta} - \widetilde{\epsilon}\mathbf{\Sigma}_{P_w}^B$. Our approach will be to upper bound $\|\mathbf{M\Delta M}\|_{\mathrm{F}}$ and $\|\mathbf{M\Sigma}_{P_w}^B\mathbf{M}\|_{\mathrm{F}}$ separately. The following intermediate result, that easily follows from the stability the of distribution G, will be useful:

Claim C.5. $0 \preceq \mathbf{\Delta} \preceq 270\gamma\mathbf{\Sigma}_{P_w}$, $\mathbf{\Sigma} - (1-\epsilon)\mathbf{\Sigma}^G_{P_w} \preceq 1.2\gamma\mathbf{\Sigma}$, *and* $\mathbf{\Sigma} \preceq (1/(1-1.2\gamma))\mathbf{\Sigma}_{P_w}$.

Proof. We start with the PSD property. We first note that

$$(1-4\epsilon)\mathbf{\Sigma}^G_{P_w} \preceq (1-\epsilon)(1-3\epsilon)\mathbf{\Sigma}^G_{P_w} \preceq \frac{\mathbf{E}_{X\sim P}[w(X)]}{\mathbf{E}_{X\sim G}[w(X)]}\mathbf{\Sigma}^G_{P_w} = \mathop{\mathbf{E}}_{X\sim G_w}[XX^\top] \preceq (1+\gamma)\mathbf{\Sigma}\,, \tag{19}$$

where the second inequality uses $w(x) \le 1$ for the denominator and the assumption that $\mathbf{E}_{X\sim P}[w(X)] \ge (1-\epsilon)\,\mathbf{E}_{X\sim G}[w(X)] \ge (1-\epsilon)(1-3\epsilon)$ for the numerator, and the last inequality uses the stability of G. Finally,

$$\mathbf{\Sigma}^G_{P_w} \preceq \frac{1+\gamma}{1-4\epsilon}\mathbf{\Sigma} \preceq \frac{1}{1-\epsilon-250\gamma+250\epsilon\gamma}\mathbf{\Sigma}\,,$$

where the first step uses (19) and the second is true for $\epsilon < \gamma/20$. The above implies that $\mathbf{\Delta} \succeq 0$.

We now show that $\mathbf{\Delta} \preceq 270\gamma\mathbf{\Sigma}_{P_w}$. First, we note that

$$\begin{aligned}\mathbf{\Sigma}_{P_w} &\succeq (1-\epsilon)\mathbf{\Sigma}^G_{P_w} = (1-\epsilon)\frac{\mathbf{E}_{X\sim G}[w(X)]}{\mathbf{E}_{X\sim P}[w(X)]}\mathop{\mathbf{E}}_{X\sim G_w}[XX^\top]\\ &\succeq (1-\epsilon)(1-3\epsilon)(1-\gamma)\mathbf{\Sigma} \succeq (1-1.2\gamma)\mathbf{\Sigma}\,,\end{aligned} \tag{20}$$

where the first inequality uses the decomposition $\mathbf{\Sigma}_{P_w} = (1-\epsilon)\mathbf{\Sigma}^G_{P_w} + \epsilon\mathbf{\Sigma}^B_{P_w}$, the second is a rewriting, the third uses stability of G along with $\mathbf{E}_{X\sim G}[w(X)] \ge 1-3\epsilon$, and the last one uses that $\epsilon < \gamma/20$. We can now complete our proof:

$$\begin{aligned}\mathbf{\Delta} &= \mathbf{\Sigma} - (1-\epsilon-250\gamma+250\gamma\epsilon)\mathbf{\Sigma}^G_{P_w}\\ &\preceq \mathbf{\Sigma} - (1-\epsilon-250\gamma+250\gamma\epsilon)(1-3\epsilon)(1-\gamma)\mathbf{\Sigma} && \text{(using middle part of (20))}\\ &\preceq \mathbf{\Sigma} - (1-\epsilon-250\gamma+250\gamma\epsilon)(1-1.15\gamma)\mathbf{\Sigma} && (\epsilon \le \gamma/20)\\ &\preceq 252\gamma\mathbf{\Sigma} && \text{(using } \epsilon \le \gamma/20 \text{ and } \gamma \le 1/20)\\ &\preceq \frac{252\gamma}{1-1.2\gamma}\mathbf{\Sigma}_{P_w} \preceq 270\gamma\mathbf{\Sigma}_{P_w}\,. && \text{(using (20) and } \gamma \le 1/20)\end{aligned}$$

The claim that $\mathbf{\Sigma} - (1-\epsilon)\mathbf{\Sigma}^G_{P_w} \preceq 1.2\gamma\mathbf{\Sigma}$ can be extracted from the middle part of (20) after some rearranging.

■

We are now ready to upper bound the variance of Y:

$$\mathbf{Var}[Y] = 2\|\mathbf{M}\mathbf{\Delta}\mathbf{M}\|^2_{\mathrm{F}} \le 4\|\mathbf{M}\mathbf{\Delta}\mathbf{M}\|^2_{\mathrm{F}} + 4\bar{\epsilon}^2\left\|\mathbf{M}\mathbf{\Sigma}^B_{P_w}\mathbf{M}\right\|^2_{\mathrm{F}}\,, \tag{21}$$

where the last inequality is triangle inequality and $(a+b)^2 \le 2a+2b$. We upper bound each term separately. Using Claim C.5, we bound the first variance term as follows:

$$\begin{aligned}\|\mathbf{M}\mathbf{\Delta}\mathbf{M}\|^2_{\mathrm{F}} &= \mathrm{tr}(\mathbf{M}^2\mathbf{\Delta}\mathbf{M}^2\mathbf{\Delta}) && \text{(using } \|\mathbf{A}\|^2_{\mathrm{F}} = \langle\mathbf{A},\mathbf{A}\rangle \text{ for symmetric } \mathbf{A} \text{ and cyclic property of trace)}\\ &\lesssim \gamma\mathrm{tr}(\mathbf{M}^2\mathbf{\Delta}\mathbf{M}^2\mathbf{\Sigma}_{P_w}) && \text{(using Claim C.5 and Fact B.3)}\\ &= \gamma\mathrm{tr}(\mathbf{M}^2\mathbf{\Sigma}_{P_w}\mathbf{M}^2\mathbf{\Delta}) && \text{(cyclic property of trace)}\\ &\lesssim \gamma^2\mathrm{tr}(\mathbf{\Sigma}_{P_w}\mathbf{M}^2\mathbf{\Sigma}_{P_w}\mathbf{M}^2) && \text{(using Claim C.5 and Fact B.3)}\\ &\le \gamma^2\|\mathbf{\Sigma}_{P_w}\|^2_{\mathrm{op}}\mathrm{tr}(\mathbf{M}^4) && \text{(using Fact B.3)}\\ &= \gamma^2\|\mathbf{\Sigma}_{P_w}\|^2_{\mathrm{op}}\|\mathbf{M}\|^4_4\\ &\le \gamma^2\|\mathbf{\Sigma}_{P_w}\|^2_{\mathrm{op}}\|\mathbf{M}\|^4_2 && \text{(using } \|\mathbf{A}\|_{q+1} \le \|\mathbf{A}\|_q)\\ &= \gamma^2\|\mathbf{\Sigma}_{P_w}\|^2_{\mathrm{op}}\|\mathbf{M}\|^4_{\mathrm{F}}\,, && (22)\end{aligned}$$

where the first application of Fact B.3 above required that $\mathbf{M}^2\mathbf{\Delta}\mathbf{M}^2 \succeq 0$ which is indeed satisfied because we have shown that $\mathbf{\Delta} \succeq 0$ in Claim C.5 and thus Fact B.5 implies $\mathbf{M}^2\mathbf{\Delta}\mathbf{M}^2 \succeq 0$. A similar argument was used in the second application of Fact B.3.

It remains to bound the second term of (21). To this end, we have that

$$\begin{aligned}\epsilon \left\langle \mathbf{\Sigma}_{P_w}^B, \mathbf{M}^2 \right\rangle &= \left\langle \mathbf{\Sigma}_{P_w} - (1-\epsilon)\mathbf{\Sigma}_{P_w}^G, \mathbf{M}^2 \right\rangle \\ &\le \left((1+O(\gamma)) \left\langle \mathbf{\Sigma}, \mathbf{M}^2 \right\rangle - (1-\epsilon) \left\langle \mathbf{\Sigma}_{P_w}^G, \mathbf{M}^2 \right\rangle\right) \\ &= \left(\left\langle \mathbf{\Sigma} - (1-\epsilon)\mathbf{\Sigma}_{P_w}^G, \mathbf{M}^2 \right\rangle + O(\gamma) \left\langle \mathbf{\Sigma}, \mathbf{M}^2 \right\rangle\right) \\ &\lesssim \gamma \left\langle \mathbf{\Sigma}, \mathbf{M}^2 \right\rangle \ , \end{aligned} \tag{23}$$

where the first line uses the decomposition $\mathbf{\Sigma}_{P_w} = (1-\epsilon)\mathbf{\Sigma}_{P_w}^G + \epsilon \mathbf{\Sigma}_{P_w}^B$, the second line uses our assumption $\langle \mathbf{\Sigma}_{P_w}, \mathbf{M}^2 \rangle \le \langle \mathbf{\Sigma}, \mathbf{M}^2 \rangle / (1-250\gamma) \le (1+O(\gamma)) \langle \mathbf{\Sigma}, \mathbf{M}^2 \rangle$, and the fourth line uses $\mathbf{\Sigma} - (1-\epsilon)\mathbf{\Sigma}_{P_w}^G \preceq 1.2\gamma \mathbf{\Sigma} \preceq 2\gamma \mathbf{\Sigma}$ by Claim C.5. Using (23), we can finally upper bound the second term of (21):

$$\begin{aligned}\tilde{\epsilon}^2 \left\| \mathbf{M} \mathbf{\Sigma}_{P_w}^B \mathbf{M} \right\|_{\mathrm{F}}^2 &\le \tilde{\epsilon}^2 \mathrm{tr}\left(\mathbf{M} \mathbf{\Sigma}_{P_w}^B \mathbf{M}\right)^2 && \text{(using Fact B.6)} \\ &= \tilde{\epsilon}^2 \left\langle \mathbf{\Sigma}_{P_w}^B, \mathbf{M}^2 \right\rangle^2 \\ &\le \epsilon^2 \left\langle \mathbf{\Sigma}_{P_w}^B, \mathbf{M}^2 \right\rangle^2 && (\tilde{\epsilon} \le \epsilon) \\ &\lesssim \gamma^2 \left\langle \mathbf{\Sigma}, \mathbf{M}^2 \right\rangle^2 && \text{(by (23))} \\ &\le \frac{\gamma^2}{1-1.2\gamma} \cdot \left\langle \mathbf{\Sigma}_{P_w}, \mathbf{M}^2 \right\rangle^2 && \text{(Claim C.5 and Fact B.3)} \\ &\lesssim \gamma^2 \|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}^2 \mathrm{tr}(\mathbf{M}^2)^2 && (24) \\ &\lesssim \gamma^2 \|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}^2 \|\mathbf{M}\|_{\mathrm{F}}^4 \ , && (25)\end{aligned}$$

where the first step uses that $\mathbf{\Sigma}_{P_w}^B$ is a PSD matrix, and thus $\mathbf{M} \mathbf{\Sigma}_{P_w}^B \mathbf{M}$ is also PSD. Putting everything together, we have shown that $\mathbf{Var}[Y] \lesssim \gamma^2 \|\mathbf{\Sigma}_{P_w}\|_{\mathrm{op}}^2 \|\mathbf{M}\|_{\mathrm{F}}^4$. ■

C.2. Reduction of Potential Function

This subsection contains the formal version of the arguments made in Sections 3.2 and 3.3. We directly give the proof of Theorem C.1, i.e., the improved runtime of $\widetilde{O}(nd/\gamma^2)$. The algorithm realizing Theorem C.1 is given in Algorithm 5.

We start by analyzing the potential $\phi_{k,t} := \mathrm{tr}(\mathbf{B}_{k,t}^{2p_k+1})$ along one iteration of the *inner loop* of Algorithm 5. That is, we fix a k of the outer loop and show that, on average, we will have $\phi_{k,t+1} \le (1-\Omega(\gamma))\phi_t$ for every step of the inner loop.

Our analysis will use that the scores $f_{k,t}(x)$ are a constant factor approximation of the scores $g_{k,t}(x)$ with constant probability. This follows from Fact 2.2. However, to make our analysis of this section more flexible and easier to adapt to the streaming setting of the next section, we will assume a weaker approximation on $f_t(\cdot)$, which includes also an additive error term (this additive error will account for approximation of $\mathbf{M}$).

Assumption C.6. The random selection of the vector $v_{k,t}$ in Line 13 of Algorithm 5 is such that for every point x,

$$\Pr_{v_{k,t}}\left[f_{k,t}(x) > \frac{g_{k,t}(x)}{10} - 0.01\frac{\gamma}{\epsilon}\|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2 \|\mathbf{\Sigma}\|_{\mathrm{op}} \right] \ge 0.4 \ .$$

We emphasize again that Fact 2.2 implies that Assumption C.6 holds for Algorithm 5.

Lemma C.7. *Consider the notation of Algorithm 5, where the input distribution is the mixture $P = (1-\epsilon)G + \epsilon B$, with G being a $(20\epsilon, \gamma)$-stable distribution with respect to $\mathbf{\Sigma}$ for $0 < 20\epsilon \le \gamma < \gamma_0$ for a sufficiently small positive constant γ_0. Make Assumption C.6. Let $\mathcal{E}_{k,t} = \mathcal{E}_{k,t}^{(1)} \cap \mathcal{E}_{k,t}^{(2)}$ denote the intersection of the following two events:*

1. *$\mathcal{E}_{k,t}^{(1)}$: $(1-\epsilon)\,\mathbf{E}_{X\sim G}[1-w_{k',t'}(X)] + \epsilon\,\mathbf{E}_{X\sim B}[w_{k',t'}(X)] \le 3\epsilon$, for every iteration (k',t') prior to (and including) (k,t).*
2. *$\mathcal{E}_{k,t}^{(2)}$: $\langle \mathbf{\Sigma}, \mathbf{M}_{k',t'}^2 \rangle < (1-250\gamma)\langle \mathbf{\Sigma}_{k',t'}, \mathbf{M}_{k',t'}^2 \rangle$ for every iteration (k',t') from $(k,1)$ up to (and including) (k,t).*

Define the potential function $\phi_{k,t} := \mathrm{tr}(\mathbf{B}_{k,t}^{2p_k+1})$. Let $F_{k,t}$ be the randomness used by HARDTHRESHOLDINGFILTER *during the (k,t)-th loop of* ROBUSTPCA *(i.e., $F_{k,t}$ is the collection of the random variables $r_1, r_2, \ldots$ used by the filter), and let $v_{k,t}$*

Algorithm 5 ROBUSTPCA with improved runtime

Input: P, ϵ, γ.
Let C, C' be sufficiently large absolute constants with their ratio C/C' being sufficiently large.
Let $k_{\mathrm{end}} := \log((\log(d/\gamma)/\log(d))/\gamma)$, and $t_{\mathrm{end}} := \frac{C\log^2(d/\epsilon)}{\gamma}$.
Find estimator $\widehat{\sigma}_{\mathrm{op}} \in (0.8\|\mathbf{\Sigma}\|_{\mathrm{op}}, 2d\|\mathbf{\Sigma}\|_{\mathrm{op}})$. ▶{c.f. Item 4 of Lemma 2.6 with $\mathbf{U} = \mathbf{I}$.}
Initialize $w_{1,1}(x) = \mathbb{1}\left(\|x\|_2^2 \le 10\widehat{\sigma}_{\mathrm{op}}(d/\epsilon)\right)$.
for $k = 1, \ldots, k_{\mathrm{end}}$ **do**
  Let $p_k = 2^{k-1}p$, where $p = C'\log(d)$. ▶{p_k ranges from $C'\log(d)$ to $C'\log(d/\gamma)/\gamma$.}
  for $t = 1, \ldots, t_{\mathrm{end}}$ **do**
    Call SAMPLETOPEIGENVECTOR $(P, w_{k,t}, \epsilon, \gamma, 1/(k_{\mathrm{end}} \cdot t_{\mathrm{end}}))$. ▶{c.f. Algorithm 2.}
    Let $P_{k,t}$ be the distribution of P weighted by $w_{k,t}$: $P(x)w_{k,t}(x)/\mathbf{E}_{X\sim P}[w_{k,t}(X)]$.
    Let $\mathbf{B}_{k,t} := \mathbf{E}_{X\sim P}[w_{k,t}(X)XX^\top]$ and $\mathbf{M}_{k,t} := \mathbf{B}_{k,t}^{p_k}$. ▶{$\mathbf{M}_{k,t}$ does not need to be explicitly computed.}
    Let $g_{k,t}(x) := \|\mathbf{M}_{k,t}x\|_2^2$.
    $v_{k,t} \leftarrow \mathbf{M}_{k,t}z_{k,t}$, where $z_{k,t} \sim \mathcal{N}(0, \mathbf{I})$.
    Let $f_{k,t}(x) = (v_{k,t}^\top x)^2$.
    Let $L_{k,t}$ be the 3ϵ-quantile of $f_{k,t}(\cdot)$ under $P_{k,t}$.
    $L_{k,t} \leftarrow \max\{L_{k,t}, (0.1/d)\widehat{\sigma}_{\mathrm{op}}\|v_{k,t}\|_2^2\}$
    Let $\tau_{k,t}(x) = f_{k,t}(x)\mathbb{1}(f_{k,t}(x) > L_{k,t})$
    Find $\widehat{\sigma}_{k,t}$ such that $|\widehat{\sigma}_{k,t} - v_{k,t}^\top\mathbf{\Sigma}v_{k,t}| \le 4\gamma v_{k,t}^\top\mathbf{\Sigma}v_{k,t}$ (e.g., $\widehat{\sigma}_{k,t} := \mathbf{E}_{X\sim P}[w_{k,t}(X)f_{k,t}(X)\mathbb{1}(f_{k,t}(X) \le L_{k,t})]$). ▶{c.f. Item 4 of Lemma 2.6 with $\mathbf{U} = v_{k,t}^\top$}
    $\widehat{T}_{k,t} \leftarrow 2.35\gamma\widehat{\sigma}_{k,t}$.
    $w_{k,t+1} \leftarrow \mathrm{HardThresholdingFilter}(P, w_{k,t}, \tau_{k,t}, \widehat{T}_{k,t}, R, 0)$. ▶{c.f. Algorithm 5.}
  end for
  Set $w_{k+1,0} \leftarrow w_{k,t+1}$.
end for
return FAIL.

be the random vectors used in Line 13 of ROBUSTPCA *(Algorithm 5). Also, define* $\mathcal{F}_{k,t} = \{F_{k',t'} : k' \le k-1, t' \le t_{\mathrm{end}}\} \cup \{F_{k,t'} : t' \le t\}$*, i.e., the entire history of filters up to (and including) the iteration* (k,t)*. Define* $\mathcal{V}_{k,t}$ *similarly for* $v_{k,t}$*'s.*

Then, the following is true for every conditioning on $\mathcal{F}_{k,t}, \mathcal{V}_{k,t-1}$*:*

$$\mathop{\mathbf{E}}_{v_{k,t}}\left[\phi_{k,t+1}\mathbb{1}(\mathcal{E}_{k,t}) \mid \mathcal{F}_{k,t}, \mathcal{V}_{k,t-1}\right] \le (1-\gamma)\phi_{k,t}\mathbb{1}(\mathcal{E}_{k,t-1})\,. \tag{26}$$

Proof. We analyze the (k,t)-th iteration of ROBUSTPCA, that is, the outer loop with index k and the inner loop with index t. We condition on everything that has happened previously, and we are only interested in analyzing what happens in the current round of the inner loop with respect to randomness coming from $v_{k,t}$. In particular, the expectation will be conditional on the past history $\mathcal{F}_{k,t}, \mathcal{V}_{k,t-1}$ as in (26) but we omit explicitly writing them for notational convenience.

In the case that $\mathbb{1}(\mathcal{E}_{k,t}) = 0$, the conclusion of our lemma holds trivially. Thus, in the reminder of the proof, we analyze the case $\mathbb{1}(\mathcal{E}_{k,t}) = 1$.

We will call a point x *full* if $f_{t,k}(x) > g_{k,t}(x)/10 - 0.01(\gamma/\epsilon)\|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2\|\mathbf{\Sigma}\|_{\mathrm{op}}$, otherwise we will call it *empty*. Note that x being full implies that

$$\begin{aligned}\tau_{k,t}(x) &\ge f_{k,t}(x) - L_{k,t} \ge \frac{g_{k,t}(x)}{10} - L_{k,t} - 0.01\frac{\gamma}{\epsilon}\|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2\|\mathbf{\Sigma}\|_{\mathrm{op}} \\ &\ge \frac{g_{k,t}(x)}{10} - 1.65\frac{\gamma}{\epsilon}\|v_{k,t}\|_2^2\|\mathbf{\Sigma}\|_{\mathrm{op}} - 0.01\frac{\gamma}{\epsilon}\|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2\|\mathbf{\Sigma}\|_{\mathrm{op}}\,,\end{aligned} \tag{27}$$

where the first inequality uses the definition of $\tau_{k,t}$, and the last inequality uses Item 3 of Lemma 2.6.

We now use Lemma B.16 to analyze the effect of filtering via HARDTHRESHOLDINGFILTER (Algorithm 4): We apply that lemma with $T = 2.35\gamma v_{k,t}^\top\mathbf{\Sigma}v_{k,t}$, $\widehat{T} = 2.35\gamma\widehat{\sigma}_{k,t}$, and $\delta = 0$; we will now justify the choice of these parameters. The first requirement of Lemma B.16 is that $(1-\epsilon)\,\mathbf{E}_{X\sim G}\left[w_{k,t}(x)\tau_{k,t}(x)\right] < T$, which is satisfied by Item 2 of Lemma 2.6

(specifically, the special case that uses $\mathbf{U} = v_{k,t}^\top$). For its second requirement of $\widehat{T} > T/5$, we have that $|\widehat{T} - T| = 2.35\gamma|\widehat{\sigma}_{k,t} - v_{k,t}^\top \mathbf{\Sigma} v_{k,t}| \le 2.35\gamma \cdot 4\gamma \cdot v_{k,t}^\top \mathbf{\Sigma} v_{k,t} \le (1/5) \cdot 2.35\gamma v_{k,t}^\top \mathbf{\Sigma} v_{k,t} = T/5$, where the second step uses Item 4 of Lemma 2.6 (with $\mathbf{U} = v_{k,t}^\top$) and the last inequality uses that $\gamma < 1/20$. Thus, the lemma is applicable, and it yields that

$$\mathop{\mathbf{E}}_{X\sim P}[w_{k,t+1}(X)\tau_{k,t}(X)] \le 3T \le 7.1\gamma\|v_{k,t}\|_2^2\|\mathbf{\Sigma}\|_{\mathrm{op}} \,, \tag{28}$$

with probability one. Therefore, for the bad points that are full, we get that their updated weights after the filter in the current iteration satisfy the following:

$$\begin{aligned}\epsilon \mathop{\mathbf{E}}_{X\sim B}[w_{k,t+1}(X)g_{k,t}(X)\mathbb{1}(X \text{ full})] &\le 10\epsilon \mathop{\mathbf{E}}_{X\sim B}[w_{k,t+1}(X)\tau_{k,t}(X)] + 16.5\gamma\|v_{k,t}\|_2^2\|\mathbf{\Sigma}\|_{\mathrm{op}} + 0.1\gamma\left\|\mathbf{M}_{k,t}\right\|_{\mathrm{F}}^2 \\ &< 88\gamma\|v_{k,t}\|_2^2\|\mathbf{\Sigma}\|_{\mathrm{op}} + 0.1\gamma\left\|\mathbf{M}_{k,t}\right\|_{\mathrm{F}}^2\|\mathbf{\Sigma}\|_{\mathrm{op}} \,. \end{aligned} \tag{29}$$

where the first step uses (27) and the second uses (28).

By Assumption C.6, every point is full with probability 0.4. We will now take expectation with respect to $v_{k,t}$ (we again remind the reader that the expectation is conditioned on the past history $\mathcal{F}_{k,t}, \mathcal{V}_{k,t-1}$ as in (26) but we omit explicitly writing them for notational convenience)).

$$\begin{aligned}&\mathop{\mathbf{E}}_{v_{k,t}}\left[\epsilon \mathop{\mathbf{E}}_{X\sim B}[w_{k,t+1}(X)g_{k,t}(X)]\right] \\ &= \mathop{\mathbf{E}}_{v_{k,t}}\left[\epsilon \mathop{\mathbf{E}}_{X\sim B}[w_{k,t+1}(X)g_{k,t}(X)\mathbb{1}(X \text{ full})]\right] + \mathop{\mathbf{E}}_{v_{k,t}}\left[\epsilon \mathop{\mathbf{E}}_{X\sim B}[w_{k,t+1}(X)g_{k,t}(X)\mathbb{1}(X \text{ empty})]\right] \\ &\le \mathop{\mathbf{E}}_{v_{k,t}}\left[\epsilon \mathop{\mathbf{E}}_{X\sim B}[w_{k,t+1}(X)g_{k,t}(X)\mathbb{1}(X \text{ full})]\right] + \epsilon \mathop{\mathbf{E}}_{X\sim B}\left[w_{k,t}(X)g_{k,t}(X)\mathop{\mathbf{E}}_{v_{k,t}}[\mathbb{1}(X \text{ empty})]\right] \\ &\le \mathop{\mathbf{E}}_{v_{k,t}}\left[\left(88\gamma\|v_{k,t}\|_2^2\|\mathbf{\Sigma}\|_{\mathrm{op}} + 0.1\gamma\left\|\mathbf{M}_{k,t}\right\|_{\mathrm{F}}^2\|\mathbf{\Sigma}\|_{\mathrm{op}}\right)\right] + \epsilon \mathop{\mathbf{E}}_{X\sim B}[w_{k,t}(X)g_{k,t}(X)(0.6)] \\ &= 88.1\gamma\|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2\|\mathbf{\Sigma}\|_{\mathrm{op}} + 0.6\epsilon \mathop{\mathbf{E}}_{X\sim B}[w_{k,t}(X)g_{k,t}(X)] \,, \end{aligned} \tag{30}$$

where the second step uses $w_{k,t+1} \le w_{k,t}$ for the second term, the third step uses (29) and the fact that the probability of a point being empty is at most 0.6,and the last step uses that $\mathbf{E}_{v_{k,t}}\left[\|v_{k,t}\|_2^2\right] = \mathbf{E}_{z_{k,t}\sim\mathcal{N}(0,\mathbf{I})}[z_{k,t}^\top \mathbf{M}_{k,t}^2 z_{k,t}] = \mathrm{tr}(\mathbf{M}_{k,t}^2) = \|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2$.

We now upper bound the expectation of the potential function using following series of inequalities, which are explained below:[15]

$$\begin{aligned}\mathop{\mathbf{E}}_{v_{k,t}}\left[\mathrm{tr}\left(\mathbf{B}_{k,t+1}^{2p_k+1}\right)\right] &\le \mathop{\mathbf{E}}_{v_{k,t}}\left[\mathrm{tr}\left(\mathbf{B}_{k,t}^{p_k}\mathbf{B}_{k,t+1}\mathbf{B}_{k,t}^{p_k}\right)\right] && (31)\\ &= \mathop{\mathbf{E}}_{v_{k,t}}\left[\mathrm{tr}\left(\mathbf{M}_{k,t}\mathbf{B}_{k,t+1}\mathbf{M}_{k,t}\right)\right] \\ &= \mathop{\mathbf{E}}_{v_{k,t}}\left[\mathop{\mathbf{E}}_{X\sim P}[w_{k,t+1}(X)g_{k,t}(X)]\right] && (32)\\ &= \mathop{\mathbf{E}}_{v_{k,t}}\left[(1-\epsilon)\mathop{\mathbf{E}}_{X\sim G}[w_{k,t+1}(X)g_{k,t}(X)] + \epsilon \mathop{\mathbf{E}}_{X\sim B}[w_{k,t+1}(X)g_{k,t}(X)]\right] && (33)\\ &\le (1-\epsilon)\mathop{\mathbf{E}}_{X\sim G}[w_{k,t}(X)g_{k,t}(X)] + \mathop{\mathbf{E}}_{v_{k,t}}\left[\epsilon \mathop{\mathbf{E}}_{X\sim B}[w_{k,t+1}(X)g_{k,t}(X)]\right] && (34)\\ &\le (1+2\gamma)\left\langle\mathbf{\Sigma}, \mathbf{M}_{k,t}^2\right\rangle + 89\gamma\|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2\|\mathbf{\Sigma}\|_{\mathrm{op}} + 0.6\epsilon \mathop{\mathbf{E}}_{X\sim B}[w_{k,t}(X)g_{k,t}(X)] && (35)\\ &= (1+2\gamma)\left\langle\mathbf{\Sigma}, \mathbf{M}_{k,t}^2\right\rangle + 89\gamma\|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2\|\mathbf{\Sigma}\|_{\mathrm{op}} \\ &\quad + 0.6\left(\mathop{\mathbf{E}}_{X\sim P}[w_{k,t}(X)g_{k,t}(X)] - (1-\epsilon)\mathop{\mathbf{E}}_{X\sim G}[w_{k,t}(X)g_{k,t}(X)]\right) && (36)\\ &= (1+2\gamma)\left\langle\mathbf{\Sigma}, \mathbf{M}_{k,t}^2\right\rangle + 89\gamma\|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2\|\mathbf{\Sigma}\|_{\mathrm{op}} + 0.6\left(\phi_{k,t} - (1-\epsilon)\mathop{\mathbf{E}}_{X\sim G}[w_{k,t}(X)g_{k,t}(X)]\right) && (37)\\ &\le (1+2\gamma)\left\langle\mathbf{\Sigma}, \mathbf{M}_{k,t}^2\right\rangle + 89\gamma\|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2\|\mathbf{\Sigma}\|_{\mathrm{op}} + 0.6\phi_{k,t} - 0.6\,(1-3\gamma)\left\langle\mathbf{\Sigma}, \mathbf{M}_{k,t}^2\right\rangle && (38)\end{aligned}$$

[15] Recall that we are in the setting when $\mathbb{1}(\mathcal{E}_{k,t-1})$=1 , and thus we omit writing the indicator inside the expectations explicitly.

$$= 0.4\langle \mathbf{\Sigma}, \mathbf{M}_{k,t}^2 \rangle + 3.8\gamma \langle \mathbf{\Sigma}, \mathbf{M}_{k,t}^2 \rangle + 89\gamma \|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2 \|\mathbf{\Sigma}\|_{\mathrm{op}} + 0.6\phi_{k,t}$$
$$\leq 0.4\langle \mathbf{\Sigma}, \mathbf{M}_{k,t}^2 \rangle + 93\gamma \|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2 \|\mathbf{\Sigma}\|_{\mathrm{op}} + 0.6\phi_{k,t} \ , \tag{39}$$

where (31) uses that $\mathbf{B}_{k,t+1} \preceq \mathbf{B}_{k,t}$ along with Fact B.4 and the cyclic property of trace operator, (32) uses the definition $\mathbf{B}_{k,t} = \mathbf{E}_{X\sim P}[w_{k,t}(X)]\mathbf{\Sigma}_{k,t} = \mathbf{E}_{X\sim P}[w_{k,t}(X)XX^\top]$, (33) uses that $P = (1-\epsilon)G + \epsilon B$, (34) uses $w_{k,t+1}(x) \leq w_{k,t}(x)$ for the first term, (35) uses $1-\epsilon \leq 1$ and Item 1 of Lemma 2.6 for the first term and (30) for the second term, (36) uses that $P = (1-\epsilon)G + \epsilon B$, (37) uses the definition of $\mathbf{B}_{k,t}$ to get $\mathbf{E}_{X\sim P}[w_{k,t}(X)g_{k,t}(X)] = \mathbf{E}_{X\sim P}[w_{k,t}(X)\mathrm{tr}(\mathbf{M}_{k,t}^2 XX^\top)] = \mathrm{tr}(\mathbf{M}_{k,t}^2\, \mathbf{E}_{X\sim P}[w_{k,t}(X)XX^\top]) = \mathrm{tr}(\mathbf{B}_{k,t}^{2p_k+1}) = \phi_{k,t}$. We now explain the last three steps. Staring with (38), recall that we have conditioned on the event that $(1-\epsilon)\,\mathbf{E}_{X\sim G}[1-w_{k,t}(X)] + \epsilon\,\mathbf{E}_{X\sim B}[w_{k,t}(X)] \leq 2.9\epsilon$. Thus, we use Item 1 of Lemma 2.6 to get $(1-\epsilon)\,\mathbf{E}_{X\sim G}[w_{k,t}(X)g_{k,t}(X)] \geq (1-\epsilon)(1-2\gamma)\langle \mathbf{\Sigma}, \mathbf{M}_{k,t}^2 \rangle \geq (1-3\gamma)\langle \mathbf{\Sigma}, \mathbf{M}_{k,t}^2 \rangle$ (where the last inequality uses $\epsilon < \gamma/20$). Finally, (39) upper bounds the terms $\gamma\langle \mathbf{\Sigma}, \mathbf{M}_{k,t}^2 \rangle$ by $\gamma\|\mathbf{\Sigma}\|_{\mathrm{op}}\|\mathbf{M}\|_{\mathrm{F}}^2$. We will now relate both $\langle \mathbf{\Sigma}, \mathbf{M}_{k,t}^2 \rangle$ and $\|\mathbf{\Sigma}\|_{\mathrm{op}}\|\mathbf{M}\|_{\mathrm{F}}^2$ with $\phi_{k,t}$.

We begin with the term $\langle \mathbf{\Sigma}, \mathbf{M}_{k,t}^2 \rangle$: Since the event $\mathcal{E}_{k,t}^{(2)}$ holds, we have that

$$\langle \mathbf{\Sigma}, \mathbf{M}_{k,t}^2 \rangle < (1-250\gamma)\langle \mathbf{\Sigma}_{k,t}, \mathbf{M}_{k,t}^2 \rangle = \frac{1-250\gamma}{\mathbf{E}_{X\sim P}[w_{k,t}(X)]}\langle \mathbf{B}_{k,t}, \mathbf{M}_{k,t}^2 \rangle \leq \frac{1-250\gamma}{(1-\epsilon)(1-3\epsilon)}\phi_{k,t} \leq (1-240\gamma)\phi_{k,t} \ , \tag{40}$$

where the first inequality uses the definition of the event $\mathcal{E}_{k,t}^{(2)}$, the next steps use that $\mathbf{E}_{X\sim P}[w_{k,t}(X)] \geq (1-\epsilon)\,\mathbf{E}_{X\sim G}[w_{k,t}(X)] \geq (1-\epsilon)(1-3\epsilon)$ (where the last inequality here used the definition of the event $\mathcal{E}_{k,t}^{(1)}$).

We now state the following result relating $\phi_{k,t}$ with $\|\mathbf{\Sigma}\|_{\mathrm{op}}\|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2$ for p_k large enough:

Claim C.8. *If $p_k \geq C\log(d)$ for a sufficiently large constant C and the event $\mathcal{E}_{k,t}$ holds, then*

$$\mathrm{tr}\left(\mathbf{B}_{k,t}^{2p_k}\right)\|\mathbf{\Sigma}\|_{\mathrm{op}} \leq 1.01 \cdot \mathrm{tr}\left(\mathbf{B}_{k,t}^{2p_k+1}\right) \ .$$

Proof. We have that

$$\begin{aligned}
\mathrm{tr}\left(\mathbf{B}_{k,t}^{2p_k}\right)\|\mathbf{\Sigma}\|_{\mathrm{op}} &< (1+3\gamma)\mathrm{tr}\left(\mathbf{B}_{k,t}^{2p_k}\right)\|\mathbf{B}_{k,t}\|_{\mathrm{op}} && \text{(using stability as in (18))} \\
&= (1+3\gamma)\|\mathbf{B}_{k,t}\|_{2p_k}^{2p_k}\|\mathbf{B}_{k,t}\|_{\infty} \\
&\leq (1+3\gamma)\|\mathbf{B}_{k,t}\|_{2p_k}^{2p_k+1} && (\|\mathbf{A}\|_{\infty} \leq \|\mathbf{A}\|_q \ \forall q) \\
&\leq (1+3\gamma)\left(d^{\frac{1}{2p_k(2p_k+1)}}\|\mathbf{B}_{k,t}\|_{2p_k+1}\right)^{2p_k+1} && \text{(by Fact 2.1)} \\
&= (1+3\gamma)d^{1/2p_k}\|\mathbf{B}_{k,t}\|_{2p_k+1}^{2p_k+1} \\
&\leq (1+3\gamma)\cdot 1.001 \cdot \mathrm{tr}\left(\mathbf{B}_{k,t}^{2p_k+1}\right) && (p_k \geq C\log(d) \text{ for } C \text{ large enough}) \\
&\leq 1.01 \cdot \mathrm{tr}\left(\mathbf{B}_{k,t}^{2p_k+1}\right) \ . && (\gamma < \gamma_0 \text{ for a sufficiently small } \gamma_0)
\end{aligned}$$

This completes the proof of Claim C.8. ■

We can now upper bound the RHS of (39) using (40) and Claim C.8:

$$\begin{aligned}
0.4\langle \mathbf{\Sigma}, \mathbf{M}_{k,t}^2 \rangle + 93\gamma\|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2\|\mathbf{\Sigma}\|_{\mathrm{op}} + 0.6\phi_{k,t} &\leq 0.4(1-240\gamma)\phi_{k,t} + 93\cdot 1.01\cdot\gamma\phi_{k,t} + 0.6\phi_{k,t} \\
&\leq (1-\gamma)\phi_{k,t} \ .
\end{aligned}$$

This completes the proof.

■

Corollary C.9. *In the context of Lemma C.7, assume that* $\mathbf{E}_{\mathcal{V}_{k-1,t_{\mathrm{end}}}}[\phi_{k,1} \mid \mathcal{F}_{k-1,t_{\mathrm{end}}}] \le R$. *Then, under every conditioning* $\mathcal{F}_{k,t-1}$ *of the form* $\mathcal{F}_{k-1,t_{\mathrm{end}}} \cup \{F_{k,t'} : t' \le t-1\}$ *(i.e., every conditioning for the filter up to the* $(k,t-1)$*-th iteration that agrees with* $\mathcal{F}_{k-1,t_{\mathrm{end}}}$ *on the first iterations up to the* $(k-1,t_{\mathrm{end}})$*-th), we have that*

$$\mathop{\mathbf{E}}_{\mathcal{V}_{k,t-1}}[\phi_{k,t}\mathbb{1}(\mathcal{E}_{k,t-1}) \mid \mathcal{F}_{k,t-1}] \le (1-\gamma)^{t-1}R\,.$$

Proof. We prove this by induction on t. The base case $t=1$ holds trivially by assumption. Now, we assume that the claim holds for the index (k,t) and we will show that it will continue to hold for the index $(k,t+1)$. That is, suppose that $\mathbf{E}_{\mathcal{V}_{k,t-1}}[\phi_{k,t}\mathbb{1}(\mathcal{E}_{k,t-1}) \mid \mathcal{F}_{k,t-1}] \le (1-\gamma)^{t-1}R$. Then, we know by Lemma C.7 that

$$\mathop{\mathbf{E}}_{v_{k,t}}[\phi_{k,t+1}\mathbb{1}(\mathcal{E}_{k,t}) \mid \mathcal{F}_{k,t}, \mathcal{V}_{k,t-1}] \le (1-\gamma)\phi_{k,t}\mathbb{1}(\mathcal{E}_{k,t-1})\,.$$

Taking expectation over $\mathcal{V}_{k,t-1}$ of both sides yields

$$\mathop{\mathbf{E}}_{\mathcal{V}_{k,t}}[\phi_{k,t+1}\mathbb{1}(\mathcal{E}_{k,t}) \mid \mathcal{F}_{k,t}] \le (1-\gamma)\mathop{\mathbf{E}}_{\mathcal{V}_{k,t-1}}[\phi_{k,t}\mathbb{1}(\mathcal{E}_{k,t-1}) \mid \mathcal{F}_{k,t-1}] \le (1-\gamma)^t R\,.$$

■

C.3. Proof of Main Theorem

We now put together the ingredients of the previous subsections to complete the proof of Theorem C.1. It suffices to show that the conclusion of the theorem holds with a small constant probability, as this probability can be boosted arbitrarily by repeating the algorithm and selecting the output u that maximizes $u^\top \mathbf{\Sigma} u$ (more precisely, an estimate of this variance that can be obtained by Item 4 of Lemma 2.6). For completeness, we give the bounds on runtime in Appendix C.3.1.

First of all, observe that if the algorithm returns a vector using SAMPLETOPEIGENVECTOR , then the resulting vector satisfies the desired guarantees with high probability. We use the same notation as in the statement of Lemma C.7. We will show that at the start of any iteration of the outer loop (i.e., for every $k \le k_{\mathrm{end}}$ and $t=1$) we have that $\mathbf{E}_{\mathcal{V}_{k-1,t_{\mathrm{end}}}}[\phi_{k,1}\mathbb{1}(\mathcal{E}^{(1)}_{k-1,t_{\mathrm{end}}})] \le (d/\epsilon)^{C_2\log d}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p_k+1}$ for a large enough positive constant C_2. We do this by induction on k. At the start of the algorithm ($k=1$), because of Line 5, every point has norm $\|x\|_2^2 \le 10(d/\epsilon)\widehat{\sigma}_{\mathrm{op}} \le (20(d^2/\epsilon))\|\mathbf{\Sigma}\|_{\mathrm{op}}$, the potential is bounded as follows

$$\phi_{1,1} \le d\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}}^{2p+1} \le \big(20(d^2/\epsilon)\|\mathbf{\Sigma}\|_{\mathrm{op}}\big)^{2p+1} \le (d/\epsilon)^{10p}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p+1} = (d/\epsilon)^{10C'\log d}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p_1+1} \le (d/\epsilon)^{C_2\log d}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p_1+1}\,.$$

Under the induction hypothesis, suppose the desired conclusion holds up to (and including) some $k \ge 1$. Let us first relate $\phi_{k,t_{\mathrm{end}}+1}$ and $\phi_{k+1,1}$ as follows:

$$\begin{aligned}\phi_{k+1,1} &= \mathrm{tr}\left(\mathbf{B}_{k,t_{\mathrm{end}}+1}^{2p_{k+1}+1}\right) = \|\mathbf{B}_{k,t_{\mathrm{end}}+1}\|_{2p_{k+1}+1}^{2p_{k+1}+1} \le \|\mathbf{B}_{k,t_{\mathrm{end}}+1}\|_{2p_k+1}^{2p_{k+1}+1} \\ &= \mathrm{tr}\left(\mathbf{B}_{k,t_{\mathrm{end}}+1}^{2p_k+1}\right)^{(2p_{k+1}+1)/(2p_k+1)} = \phi_{k,t_{\mathrm{end}+1}}^{(2p_{k+1}+1)/(2p_k+1)}\,. \end{aligned} \tag{41}$$

By induction hypothesis, at the beginning of the k-th iteration of the outer loop, it holds that $\mathbf{E}_{\mathcal{V}_{k-1,t_{\mathrm{end}}}}[\phi_{k,1}\mathbb{1}(\mathcal{E}^{(1)}_{k-1,t_{\mathrm{end}}})] \le (d/\epsilon)^{C_2\log d}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p_k+1}$. In the k-th iteration of the outerloop, we will show that the end of the inner loop (i.e., at $t = t_{\mathrm{end}}+1$), the potential will be much smaller. In particular, we will show that $\mathbf{E}_{\mathcal{V}_{k,t_{\mathrm{end}}}}\left[\phi_{k,t_{\mathrm{end}}+1}\mathbb{1}(\mathcal{E}^{(1)}_{k,t_{\mathrm{end}}})\right] \le (d/\epsilon)^{0.1C_2\log d}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p_{k-1}+1}$ below. Note that in the k-th iteration of the algorithm, two cases might happen: (i) either the condition $\langle \mathbf{\Sigma}_t, \mathbf{M}_{k,t}^2\rangle < (1-250\gamma)\langle \mathbf{\Sigma}_{k,t}, \mathbf{M}_{k,t}^2\rangle$ holds for all t_{end} iterations of the inner loop, which would lead to a $(1-\gamma)^{t_{\mathrm{end}}}$ decrease in potential, or this condition fails to hold for some t, which itself implies that the potential is small. Recall that $\mathcal{E}^{(2)}_{k,t}$ corresponds to these two cases. Thus, we have the following decomposition:

$$\mathop{\mathbf{E}}_{\mathcal{V}_{k,t_{\mathrm{end}}}}\left[\phi_{k,t_{\mathrm{end}}+1}\mathbb{1}(\mathcal{E}^{(1)}_{k,t_{\mathrm{end}}})\right] = \mathop{\mathbf{E}}_{\mathcal{V}_{k,t_{\mathrm{end}}}}\left[\phi_{k,t_{\mathrm{end}}+1}\mathbb{1}(\mathcal{E}^{(1)}_{k,t_{\mathrm{end}}})\mathbb{1}(\mathcal{E}^{(2)}_{k,t_{\mathrm{end}}})\right] + \mathop{\mathbf{E}}_{\mathcal{V}_{k,t_{\mathrm{end}}}}\left[\phi_{k,t_{\mathrm{end}}+1}\mathbb{1}(\mathcal{E}^{(1)}_{k,t_{\mathrm{end}}})\mathbb{1}\left(\overline{\mathcal{E}^{(2)}_{k,t_{\mathrm{end}}}}\right)\right] \tag{42}$$

The first term in (42) van be bounded using Corollary C.9 as follows:

$$\mathop{\mathbf{E}}_{\mathcal{V}_{k,t_{\mathrm{end}}}}\left[\phi_{k,t_{\mathrm{end}}+1}\mathbb{1}(\mathcal{E}^{(1)}_{k,t_{\mathrm{end}}})\mathbb{1}(\mathcal{E}^{(2)}_{k,t_{\mathrm{end}}})\right] \le (1-\gamma)^{t_{\mathrm{end}}+1}(d/\epsilon)^{C_2\log d}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p_k+1} \le e^{-0.1\cdot t_{\mathrm{end}}\cdot\gamma}(d/\epsilon)^{C_2\log d}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p_k+1}, \tag{43}$$

which is less than $(d/\epsilon)^{0.05C_2\log d}$ after $t_{\mathrm{end}} := C\log^2(d/\epsilon)/\gamma$ for sufficiently large C (note that we have picked C to be much larger than C_2).

We now turn our attention to the second term in (42), where we show that if $\mathcal{E}^{(2)}_{k,t_{\mathrm{end}}}$ does not hold, then the potential will be $\mathrm{poly}(d/\epsilon)\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p_k+1}$. More formally, if for some (k,t) we have that $\langle\mathbf{\Sigma},\mathbf{M}_{k,t}^2\rangle \ge (1-250\gamma)\langle\mathbf{\Sigma}_{k,t},\mathbf{M}_{k,t}^2\rangle$, there exists c such that

$$\begin{aligned}
\phi_{k,t} = \|\mathbf{B}_{k,t}\|_{2p_k+1}^{2p_k+1} = \langle\mathbf{B}_{k,t},\mathbf{M}_{k,t}^2\rangle &\le \langle\mathbf{\Sigma}_{k,t},\mathbf{M}_{k,t}^2\rangle \le \frac{1}{1-250\gamma}\langle\mathbf{\Sigma},\mathbf{M}_{k,t}^2\rangle \\
&\le e^{c\gamma}\|\mathbf{\Sigma}\|_{\mathrm{op}}\|\mathbf{B}_{k,t}\|_{2p_k}^{2p_k} && \text{(using } \gamma \ll 1\text{)} \\
&\le e^{c\gamma}\|\mathbf{\Sigma}\|_{\mathrm{op}}\|\mathbf{B}_{k,t}\|_{2p_k+1}^{2p_k} d^{\frac{1}{2p_k+1}} . && \text{(using Fact 2.1)}
\end{aligned}$$

Rearranging the inequality above implies that $\|\mathbf{B}_{k,t}\|_{2p_k+1}^{2p_k+1} \le e^{c\gamma p_k}d\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p_k+1} \le (d/\gamma)^{O(1)}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p_k+1}$, where the last inequality used that $p_k = O(\log(d/\gamma)/\gamma)$ for all k. Combining this with (42) and (43), we obtain that $\mathbf{E}_{\mathcal{V}_{k,t_{\mathrm{end}}}}\left[\phi_{k,t_{\mathrm{end}}+1}\mathbb{1}(\mathcal{E}^{(1)}_{k,t_{\mathrm{end}}})\right] \le (d/\epsilon)^{0.1C_2\log d}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p_k+1}$. Combining this with (41) and the observation that $(2p_{k+1}+1)/(2p_k+1) \le 2$ by definition of p_k (Line 8 in Algorithm 5), we conclude that

$$\mathop{\mathbf{E}}_{\mathcal{V}_{k,t_{\mathrm{end}}}}\left[\phi_{k+1,1}\mathbb{1}(\mathcal{E}^{(1)}_{k,t_{\mathrm{end}}})\right] \le \|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p_{k+1}+1}(d/\epsilon)^{C_2\log d} .$$

Thus far, we have shown that $\mathbf{E}_{\mathcal{V}_{k-1,t_{\mathrm{end}}}}[\phi_{k,1}\mathbb{1}(\mathcal{E}^{(1)}_{k-1,t_{\mathrm{end}}})] \le \|\mathbf{\Sigma}\|_{\mathrm{op}}^{2p_k+1}(d/\epsilon)^{C_2\log d}$ for $k = 1,\dots,k_{\mathrm{end}}$.

It remains to analyze the last iteration $k = k_{\mathrm{end}}$. For that, we again use Corollary C.9 to obtain the following:

$$\begin{aligned}
\mathop{\mathbf{E}}_{\mathcal{V}_{k_{\mathrm{end}},t_{\mathrm{end}}}}[\phi_{k_{\mathrm{end}},t_{\mathrm{end}}+1}\mathbb{1}(\mathcal{E}_{k_{\mathrm{end}},t_{\mathrm{end}}})] &\le (1-\gamma)^{t_{\mathrm{end}}+1}(d/\epsilon)^{C_2\log d}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{1+2p_{k_{\mathrm{end}}}} \\
&\le (d/\epsilon)^{-100C'}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{1+2p_{k_{\mathrm{end}}}} ,
\end{aligned} \tag{44}$$

where the last step uses that $t_{\mathrm{end}} := C\log^2(d/\epsilon)/\gamma$ for sufficiently large C and recalling that the constant C is large relative to the constant C' (c.f. Line 2).

Recall the definition of the event $\mathcal{E}_{k,t}$ given in the statement of Lemma C.7 as intersection of the two events:

1. $\mathcal{E}^{(1)}_{k,t}$: $(1-\epsilon)\,\mathbf{E}_{X\sim G}[1-w_{k',t'}(X)]+\epsilon\,\mathbf{E}_{X\sim B}[w_{k',t'}(X)] \le 3\epsilon$, for every iteration (k',t') prior to (and including) (k,t).
2. $\mathcal{E}^{(2)}_{k,t}$: $\langle\mathbf{\Sigma},\mathbf{M}_{k',t'}^2\rangle < (1-250\gamma)\langle\mathbf{\Sigma}_{k',t'},\mathbf{M}_{k',t'}^2\rangle$ for every iteration (k',t') from $(k,1)$ up to (and including) (k,t).

By Lemma B.17 we have that $\mathbf{Pr}[\mathcal{E}^{(1)}_{k_{\mathrm{end}},t_{\mathrm{end}}}] > 0.2$. Thus

$$\mathop{\mathbf{E}}_{\mathcal{V}_{k,t_{\mathrm{end}}}}\left[\phi_{k_{\mathrm{end}},t_{\mathrm{end}}+1}\mathbb{1}(\mathcal{E}^{(2)}_{k_{\mathrm{end}},t_{\mathrm{end}}}) \mid \mathcal{E}^{(1)}_{k_{\mathrm{end}},t_{\mathrm{end}}}\right] = \frac{\mathbf{E}_{\mathcal{V}_{k,t_{\mathrm{end}}}}[\phi_{k_{\mathrm{end}},t_{\mathrm{end}}+1}\mathbb{1}(\mathcal{E}_{k_{\mathrm{end}},t_{\mathrm{end}}})]}{\mathbf{Pr}[\mathcal{E}^{(1)}_{k_{\mathrm{end}},t_{\mathrm{end}}}]} \le 5(d/\epsilon)^{-100C'}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{1+2p_{k_{\mathrm{end}}}}$$

Therefore, we have that

$$\begin{aligned}
&\mathbf{Pr}\left[\phi_{k_{\mathrm{end}},t_{\mathrm{end}}+1}\mathbb{1}(\mathcal{E}^{(2)}_{k_{\mathrm{end}},t_{\mathrm{end}}}) > 500(d/\epsilon)^{-100C'}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{1+2p_{k_{\mathrm{end}}}} \text{ or } \mathbb{1}(\mathcal{E}^{(1)}_{k_{\mathrm{end}},t_{\mathrm{end}}}) = 0\right] \\
&\le \mathbf{Pr}[\mathbb{1}(\mathcal{E}^{(1)}_{k_{\mathrm{end}},t_{\mathrm{end}}}) = 0] + \mathbf{Pr}\left[\phi_{k_{\mathrm{end}},t_{\mathrm{end}}+1}\mathbb{1}(\mathcal{E}^{(2)}_{k_{\mathrm{end}},t_{\mathrm{end}}}) > 500(d/\epsilon)^{-100C'}\|\mathbf{\Sigma}\|_{\mathrm{op}}^{1+2p_{k_{\mathrm{end}}}} \mid \mathcal{E}^{(1)}_{k_{\mathrm{end}},t_{\mathrm{end}}}\right] \\
&\le 0.2 + 1/100 ,
\end{aligned} \tag{45}$$

where the last step uses Markov's inequality.

Let t^* be the first time during the k_{end}-th outer loop iteration of Algorithm 5 such that $\langle \mathbf{\Sigma}, \mathbf{M}^2_{k_{\text{end}},t} \rangle \geq (1 - 250\gamma)\langle \mathbf{\Sigma}_{k_{\text{end}},t}, \mathbf{M}^2_{k_{\text{end}},t} \rangle$ (or equivalently $\mathbb{1}(\mathcal{E}^{(2)}_{k_{\text{end}},t}) = 0$). In the following, we argue that $t^* \leq t_{\text{end}}$. We prove this by contradiction. Assume that $\mathbb{1}(\mathcal{E}^{(2)}_{k_{\text{end}},t_{\text{end}}}) = 1$.

In the event that $\phi_{k_{\text{end}},t_{\text{end}}+1}\mathbb{1}(\mathcal{E}^{(2)}_{k_{\text{end}},t_{\text{end}}}) < 500(d/\epsilon)^{-100C'}$ and $\mathbb{1}(\mathcal{E}^{(1)}_{k_{\text{end}},t_{\text{end}}}) = 1$ simultaneously (by (45) the two events will hold simultaneously with probability at least 0.79), we have that

$$\begin{aligned}
\phi_{k_{\text{end}},t_{\text{end}}+1} &= \phi_{k_{\text{end}},t_{\text{end}}+1}\mathbb{1}(\mathcal{E}^{(2)}_{k_{\text{end}},t_{\text{end}}}) \\
&< 500(d/\epsilon)^{-100C'} \|\mathbf{\Sigma}\|_{\text{op}}^{2p_{k_{\text{end}}}+1} \\
&< 0.5(1-2\gamma)^{2p_{k_{\text{end}}}} \|\mathbf{\Sigma}\|_{\text{op}}^{2p_{k_{\text{end}}}+1} \\
&\leq \mathop{\mathbf{E}}_{X\sim P}[w_{k,t}(X)](1-2\gamma)^{2p_{k_{\text{end}}}} \frac{1}{1-250\gamma} \|\mathbf{\Sigma}\|_{\text{op}}^{2p_{k_{\text{end}}}+1} \ , \qquad (46)
\end{aligned}$$

where the first line uses the assumption $\mathbb{1}(\mathcal{E}^{(2)}_{k_{\text{end}},t_{\text{end}}}) = 1$, the third line uses $p_{k_{\text{end}}} = C' \log(d/\gamma)/\gamma$, and the fourth line uses that $\mathbf{E}_{X\sim P}[w_{k,t}(X)] \geq 1/2$ because the event $\mathcal{E}^{(1)}_{k_{\text{end}},t_{\text{end}}}$ holds.

Lemma C.4 and (46) imply that $\langle \mathbf{\Sigma}, \mathbf{M}^2_{k_{\text{end}},t} \rangle \geq (1 - 250\gamma)\langle \mathbf{\Sigma}_{k_{\text{end}},t}, \mathbf{M}^2_{k_{\text{end}},t} \rangle$, which yields a contradiction. Therefore, $t^* \leq t_{\text{end}}$.

To complete the proof it remains to show that, if the algorithm hasn't terminated earlier with a good solution, then, during the (k_{end}, t^*)-th iteration, the algorithm will return a good approximation of the top eigenvalue of $\mathbf{\Sigma}$ and terminate. We use Lemma C.2 for that (the lemma is applicable since its requirement $\mathbf{E}_{X\sim G}[w_{k,t}(X)] \geq 1 - 3\epsilon$ follows by the fact that we have conditioned on the event $\mathcal{E}_{k_{\text{end}},t_{\text{end}}}$, which as we saw earlier happens with constant probability). The conclusion of the lemma implies that there is probability 0.9 that when $t = t^*$, the output of SAMPLETOPEIGENVECTOR satisfies $u^\top \mathbf{\Sigma} u > (1 - O(\gamma))u^\top \mathbf{\Sigma}_{k,t} u$ and $\frac{u^\top \mathbf{\Sigma}_{k,t} u}{\|u\|_2^2} \geq (1-\gamma)\|\mathbf{\Sigma}_{k,t}\|_{\text{op}}$. Conditioning on this event and by noting that the estimator $\widehat{r}_t$ of Line 3 will satisfy $\widehat{r}_t \geq (1-\gamma)\|\mathbf{\Sigma}\|_{\text{op}}$ with high probability over all the iterations, the check of Line 7 will be activated, and by Lemma B.15, the algorithm will return a vector u such that $u^\top \mathbf{\Sigma} u/\|u\|_2^2 \geq (1 - O(\gamma))\|\mathbf{\Sigma}\|_{\text{op}}$.

C.3.1. RUNTIME ANALYSIS

Let the input distribution P be the uniform distribution over n points in $\mathbb{R}^d$. The outer loop is repeated k_{end} times and the inner loop is repeated t_{end} times, where $k_{\text{end}} = O(\log(1/\gamma))$, and $t_{\text{end}} = O(\log^2(d/\epsilon)/\gamma)$.

Inside each loop, the runtime is determined by the following: Calculating $v_{k,t}$ in 13 can be implemented in time $O(ndp_k)$ by starting with $z_{k,t}$ and repeatedly multiplying it by $\mathbf{B}_{k,t}$ (multiplication of a vector z with a second moment matrix $\sum_x xx^\top$ can be implemented in $O(nd)$ time by first calculating the inner product inside the parenthesis $\sum_x x(x^\top z)$, and then calculating the average of the resulting vectors). The power iteration estimator of Line 3 in Algorithm 2 runs in time $O(\frac{nd}{\gamma}\log(d \cdot k_{\text{end}} \cdot t_{\text{end}}/\gamma))$ (and is being called at most $k_{\text{end}} \cdot t_{\text{end}}$ many times). It remains to analyze the runtime of HARDTHRESHOLDINGFILTER (Algorithm 4): We have that the scores $\tau_{k,t}(x)$ that are not zeroed out by the thresholding satisfy the following upper and lower bound: $\text{poly}(\epsilon/d)\|\mathbf{\Sigma}\|_{\text{op}}\|v_{k,t}\|_2^2 \leq 0.1(\widehat{\sigma}_{\text{op}}/d)\|v_{k,t}\|_2^2 \leq \tau_{k,t}(x) \leq \|x\|_2^2\|v_{k,t}\|_2^2 \leq 2\widehat{\sigma}_{\text{op}}(d^4/\epsilon)\|v_{k,t}\|_2^2 \leq \text{poly}(d/\epsilon)\|\mathbf{\Sigma}\|_{\text{op}}\|v_{k,t}\|_2^2$, where we used the pruning of Line 5, the definition of Line 16, and the naïve estimator of Line 6. Since the HARDTHRESHOLDINGFILTER in expectation halves the maximum value of $\tau(x)$, the number of filtering steps it performs before terminating will be in expectation $O(\log(d/\epsilon))$, and thus by Markov's inequality, the contribution to the runtime from all the $k_{\text{end}} \cdot t_{\text{end}}$ executions of HARDTHRESHOLDINGFILTER will be $O(nd \cdot k_{\text{end}} t_{\text{end}} \log(d/\epsilon))$ with high constant probability.

Therefore, the total runtime is

$$\begin{aligned}
T &= O\left(\frac{nd}{\gamma} k_{\text{end}} t_{\text{end}} \log(d \cdot k_{\text{end}} \cdot t_{\text{end}}/\gamma)\right) + O(nd \cdot k_{\text{end}} t_{\text{end}} \log(d/\epsilon)) + O\left(nd \cdot t_{\text{end}} \sum_{k=1}^{k_{\text{end}}} p_k\right) \\
&= O\left(\frac{nd}{\gamma^2} \log(1/\gamma) \log^2(d/\epsilon) \log\left(\frac{d}{\gamma}\log(d/\epsilon)\right) + \frac{nd}{\gamma}\log(1/\gamma)\log^3(d/\epsilon) + \frac{nd}{\gamma^2}\log^2(d/\epsilon)\log(d/\gamma)\right)
\end{aligned}$$

$$= O\left(\frac{nd}{\gamma^2}\log^4(d/\epsilon)\right) .$$

D. Robust PCA in Streaming Setting

We now switch to the streaming setting. We use the standard single-pass streaming model, where instead of having a fixed dataset, the algorithm draws samples from the distribution in an online manner. Let G be an $(20\epsilon, \gamma)$-stable distribution, which will be the underlying distribution of inliers. Let the "contaminated" distribution be P, which is assumed to be an ϵ-corrupted version of G in total variation (c.f. Definition 1.4). Note that up to some small change in the constant in front of ϵ, we can equivalently use a mixture representation $P = (1-\epsilon)G + \epsilon B$ (see discussion below Definition 2.4). In contrast to the previous section, where the input dataset was stored in essentially "free" memory, now the data access model is the following:

Definition D.1 (Single-Pass Streaming Model)**.** *Let P be a distribution on $\mathbb{R}^d$. First, the algorithm specifies a number n, then a set S of n samples are drawn i.i.d. from the distribution P. Finally, the elements of S are revealed one at a time to the algorithm, and the algorithm is allowed a single pass over these points.*

The algorithm is still allowed to maintain a local memory, but anything stored in the local memory will be counted against its space complexity. The goal is again to find a unit vector u such that $u^\top \mathbf{\Sigma} u \geq (1 - O(\gamma))\|\mathbf{\Sigma}\|_{\mathrm{op}}$.

Theorem D.2. *Let an integer $d > 2$, and reals $0 < 20\epsilon < \gamma < \gamma_0$, for a sufficiently small γ_0. Let G be $(20\epsilon, \gamma)$-stable distribution (Definition 2.4) with respect to a PSD matrix $\mathbf{\Sigma} \in \mathbb{R}^{d\times d}$, and r be a radius such that $\mathbf{Pr}_{X\sim G}[\|X\|_2 > r\sqrt{d\|\mathbf{\Sigma}\|_{\mathrm{op}}}] \leq \epsilon$. Let P be a distribution with $d_{\mathrm{TV}}(P, G) \leq \epsilon$. There exists an algorithm takes ϵ, γ, r as input, uses a stream of*

$$n \lesssim \left(\frac{r^2 d^2}{\gamma^5} + \frac{1}{\epsilon\gamma}\right)\mathrm{polylog}(d/\epsilon) .$$

i.i.d samples from P (cf. Definition D.1), uses additional memory of storing $O\left((d/\gamma + 1/\epsilon)\mathrm{polylog}(d/\epsilon)\right)$ many real numbers (or a bit complexity of $(d/\gamma^2)\mathrm{polylog}(d/\epsilon)$ in the word RAM Model), runs for $O(\frac{nd}{\gamma^2}\mathrm{polylog}(d/\epsilon))$ time, and with probability at least 0.99 outputs a vector u such that $u^\top \mathbf{\Sigma} u \geq (1 - O(\gamma))\|\mathbf{\Sigma}\|_{\mathrm{op}}$.

The specialization of the above theorem for subgaussians, where $\gamma = O(\epsilon\log(1/\epsilon))$ and $r = O(\sqrt{\log(1/\epsilon)})$ yields sample complexity $O((d^2/\epsilon^5)\mathrm{polylog}(d/\epsilon))$. Moreover, the memory usage stated in Theorem D.2 is in terms of the number of *real numbers* that need to be stored in the algorithm's memory. See Appendix D.3.1 for how our algorithm can be implemented with bounded precision (i.e., in the standard word RAM model) using $(d/\gamma)\mathrm{polylog}(d/\epsilon)$ registers of size $(1/\gamma)\mathrm{polylog}(d/\epsilon)$ bits each (for a total bit complexity of $(d/\gamma^2)\mathrm{polylog}(d/\epsilon)$.

Key Differences in Streaming Setting The algorithm from the previous section is partly already amenable to the streaming setting: The filters used are of the form $\mathbb{1}(v^\top x > L)$ which have a compact representation of $O(d)$ space (it suffices to store the vector v and the threshold L). The potential-based analysis also showed that we create at most $O(\mathrm{polylog}(d/\epsilon)/\gamma)$ many such filters. The remaining adaptation that needs to be done is to deal with the fact that there is no fixed dataset to iterate over. The new algorithm is given in Algorithm 7. Regarding notation, the quantities $P_{k,t}, \mathbf{\Sigma}_{k,t}, \mathbf{B}_{k,t}, \mathbf{M}_{k,t}$ as well as the score functions $g_{k,t}(x) = \|\mathbf{M}_{k,t}x\|_2^2, f_{k,t}(x) = (v_{k,t}^\top x)^2, \tau_{k,t}(x) = f_{k,t}(x)\mathbb{1}(f_{k,t}(x) > L_{k,t})$ are all population-level quantities, i.e., they are unknown to the algorithm. However, the algorithm can approximate them by drawing samples and forming estimators (where the exact form of the estimators will have to be easily computable in the streaming setting and will be discussed later on). We will use "hat" to denote the relevant sample-based approximations, i.e., $\widehat{\mathbf{\Sigma}}_{k,t}, \widehat{\mathbf{B}}_{k,t}, \widehat{\mathbf{M}}_{k,t}$ and the induced scores $\widehat{g}_{k,t}(x) = \|\widehat{\mathbf{M}}_{k,t}x\|_2^2, \widehat{f}_{k,t}(x) = (\widehat{v}_{k,t}^\top x)^2, \widehat{\tau}_{k,t}(x) = \widehat{f}_{k,t}(x)\mathbb{1}(\widehat{f}_{k,t}(x) > \widehat{L}_{k,t})$. The approach that we follow is that if the sample-based quantities are sufficiently close to their population-level counterparts, then the proof of correctness from the previous section (which involves the population-level quantities) still applies. One can easily go through the proofs of the previous section and verify that there is enough slack in all bounds to allow for converting between population-level quantities and their sample-based approximations. In this section, we avoid repeating all the correctness proofs since the only changes will be in the constants used in some inequalities, instead we mostly focus on deriving the sample complexity needed for obtaining fine enough approximations.

Sample-based estimators Many steps of the algorithm of the previous section involved multiplying a vector x by $\mathbf{M}_{k,t} := \mathbf{B}_{k,t}^{p_k}$, for example the scores $g_{k,t}(x)$ involve the norm of $\mathbf{M}_{k,t}x$. This was done by starting with x and repeatedly

multiplying by $\mathbf{B}_{k,t}$. Instead of $\mathbf{B}_{k,t}$ the new algorithm will now use an empirical moment matrix $\widehat{\mathbf{B}}_{k,t}$. As we have seen in the previous section, the multiplication of an empirical second moment matrix with a vector can be performed with $O(d)$ memory usage in a single pass over the samples and in linear time. However, since the algorithm cannot store this matrix, it will repeatedly multiply x by fresh estimators $\widehat{\mathbf{B}}_{k,t,1}, \ldots, \widehat{\mathbf{B}}_{k,t,p_k}$ each time. Thus, the result will be of the form $\widehat{\mathbf{M}}_{k,t} z$ where $\widehat{\mathbf{M}}_{k,t} = \prod_{\ell=1}^{p_k} \widehat{\mathbf{B}}_{k,t,\ell}$ (also see Algorithm 6 for more detail).

Algorithm 6 Estimator of $\mathbf{B}_{k,t}^p$ from minibatches.

1: Draw a batch S_0 of $\tilde{n}$ samples from P and let the estimate $\widehat{W}_{k,t} = \mathbf{E}_{X \sim \mathcal{U}(S_0)}[w_{k,t}(X)]$.
2: Draw p batches $S_1, \ldots, S_p$ of $\tilde{n}$ samples, each from $P_{k,t}$.
3: **for** $\ell \in [p]$ **do**
4: Let $\widehat{\mathbf{\Sigma}}_{k,t,\ell} = \frac{1}{\tilde{n}} \sum_{x \in S_\ell} x x^T$.
5: Let $\widehat{\mathbf{B}}_{k,t,\ell} = \widehat{W}_{k,t}^2 \widehat{\mathbf{\Sigma}}_{k,t,\ell}$.
6: **end for**
7: **return** $\widehat{\mathbf{M}}_{k,t,\ell} = \prod_{\ell=1}^{p} \widehat{\mathbf{B}}_{k,t,\ell}$.

Another kind of estimator that is needed in this section is an estimator for the quantiles of the underlying distribution to replace Line 15 of the old algorithm. These will be computed by empirical quantiles. Finally, another approximation needs to take place when evaluating the stopping condition inside the HARDTHRESHOLDINGFILTER (Algorithm 4) since the $\mathbf{E}_{X \sim P}[w(x)\tau(x)]$ used earlier is a population-level quantity. A similar estimator needs to be used when adapting line 18 of our old algorithm. This completes the short informal overview.

Formally, we aggregate all the guarantees needed regarding the estimators in Condition D.3. As a small technical note, the conditions in Condition D.3 are only needed to hold whenever $\mathbf{E}_{X \sim P}[w_{k,t}(X)] \geq 1 - 3\epsilon$ and $\|\mathbf{B}_{k,t}\|_{\mathrm{op}} \geq 0.5\|\mathbf{\Sigma}\|_{\mathrm{op}}$. The first is because as we filter out mostly outliers, we have already seen that $\mathbf{E}_{X \sim P}[w_{k,t}(X)] \geq 1 - 3\epsilon$ with high probability throughout the algorithm. The second is because, we also know that if $\|\mathbf{B}_{k,t}\|_{\mathrm{op}} \geq 0.5\|\mathbf{\Sigma}\|_{\mathrm{op}}$ is violated then the potential function is small enough so that the algorithm must had already terminated (c.f. Lemma C.4) holds with high probability thought the algorithm and dedicate Appendix D.1 to establishing it.

Condition D.3 (Conditions for Algorithm 7)**.** In the context of Algorithm 7, assume that the following are true. For every $k \in [k_{\mathrm{end}}]$ and $t \in [t_{\mathrm{end}}]$, if $\mathbf{E}_{X \sim P}[w_{k,t}(X)] \geq 1 - 3\epsilon$ and $\|\mathbf{B}_{k,t}\|_{\mathrm{op}} \geq 0.5\|\mathbf{\Sigma}\|_{\mathrm{op}}$, then:

1. $\widehat{g}_{k,t}(x) \geq 0.5 g_{k,t}(x) - 0.01(\gamma/\epsilon)\|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2 \|\mathbf{\Sigma}\|_{\mathrm{op}}$.
2. $\left| \|\widehat{\mathbf{M}}_{k,t}\|_{\mathrm{F}}^2 - \|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2 \right| \leq 0.01\|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2$.
3. The estimator $\widehat{L}_{k,t}$ of Line 16 satisfies $\mathbf{Pr}_{X \sim P_{k,t}}[\widehat{g}_{k,t}(X) > \widehat{L}_{k,t}] \in (3.99\epsilon, 4.01\epsilon)$.
4. Recall the parameter r as the radius such that $\mathbf{Pr}_{X \sim G}[\|X\|_2 > r\sqrt{d\|\mathbf{\Sigma}\|_{\mathrm{op}}}] \leq \epsilon$. For any weight function $w : \mathbb{R}^d \to [0,1]$, the algorithm has access to an estimator $\widehat{F}$ for the quantity $F_{k,t} := \mathbf{E}_{X \sim P}[w(X)\widehat{f}_{k,t}(X)]$ that has accuracy $|\widehat{F} - F_{k,t}| \leq 0.01\gamma F_{k,t} + \frac{0.01\gamma}{dr^2}\|\widehat{v}_{k,t}\|_2^2 \|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}}$ across a total of $t_{\mathrm{end}} k_{\mathrm{end}} \cdot \log^5(d/\epsilon)$ calls.
5. The estimator $\widehat{r}_t$ of Line 7 of Algorithm 8 satisfies $\widehat{r}_t \geq (1-\gamma)\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}}$.
6. Every time Line 10 of of Algorithm 8 is executed, there is probability 0.9 that $u^\top \mathbf{\Sigma}_{k,t} u / \|u\|_2^2 \geq (1-\gamma)\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}}$.

In particular, Item 1 of Condition D.3 shows that Assumption C.6 that we used in proving the main theorem of the previous section is satisfied for Algorithm 7 in our current setting. Item 2 is relevant to equation (27) in our previous analysis. To adapt that in our current setting, $v_{k,t}$ will be replaced by $\widehat{v}_{k,t}$ and after we take expectation over $\widehat{v}_{k,t}$ it's norm will become $\|\widehat{\mathbf{M}}_{k,t}\|_{\mathrm{F}}^2$ that we can relate to $\|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2$. Item 3 takes care of the quantile estimation. Item 4 is the estimator that we will use to evalueate the stopping condition inside HARDTHRESHOLDINGFILTER as well as adapting line 18 of the old algorithm. Items 5 and 6 are the adaptation of the power iteration guarantee when we do not have access to the target matrix $\mathbf{\Sigma}_{k,t}$ but instead we start with a random Gaussian vector and multiply it repetitively with fresh estimates $\widehat{\mathbf{\Sigma}}_{k,t}$.

Algorithm 7 ROBUSTPCA in streaming model

Input: P, ϵ, γ.
Let C, C' be sufficiently large absolute constants with their ratio C/C' being sufficiently large.
Let an estimation R be such that $|\mathbf{Pr}_{X\sim G}[\|X\|_2 \geq R] - \epsilon| \leq 2\epsilon$ (note that $R \leq r\sqrt{d\|\mathbf{\Sigma}\|_{\mathrm{op}}}$)
Initialize $w_{1,1}(x) = \mathbb{1}\left(\|x\|_2 \leq R\right)$.
Let $k_{\mathrm{end}} := \log((\log(d/\gamma)/\log(d))/\gamma)$, and $t_{\mathrm{end}} := \frac{C\log^2(d/\epsilon)}{\gamma}$.
Find estimator $\widehat{\sigma}_{\mathrm{op}} \in (0.8\|\mathbf{\Sigma}\|_{\mathrm{op}}, 2d\|\mathbf{\Sigma}\|_{\mathrm{op}})$. ▶{c.f. Item 4 of Lemma 2.6 with $\mathbf{U} = \mathbf{I}$.}
for $k = 1, \ldots, k_{\mathrm{end}}$ **do**
  Let $p_k = 2^{k-1}p$, where $p = C'\log(d)$. ▶{p_k ranges from $C'\log(d)$ to $C'\log(d/\gamma)/\gamma$.}
  for $t = 1, \ldots, t_{\mathrm{end}}$ **do**
    Let $P_{k,t}$ be the distribution of P weighted by $w_{k,t}$: $P(x)w_{k,t}(x)/\mathbf{E}_{X\sim P}[w_{k,t}(X)]$.
    Let $\mathbf{B}_{k,t} := \mathbf{E}_{X\sim P}[w_{k,t}(X)XX^\top]$ and $\mathbf{M}_{k,t} := \mathbf{B}_{k,t}^{p_k}$.
    Let $g_{k,t}(x) := \|\mathbf{M}_{k,t}x\|_2^2$.
    Let $\widehat{\mathbf{M}}_{k,t}$ be a sample-based version of $\mathbf{M}_{k,t}$ as defined in Algorithm 6.
    $\widehat{v}_{k,t} \leftarrow \widehat{\mathbf{M}}_{k,t}z_{k,t}$, where $z_{k,t} \sim \mathcal{N}(0, \mathbf{I})$.
    Let $\widehat{f}_{k,t}(x) = (\widehat{v}_{k,t}^\top x)^2$.
    Let $\widehat{L}_{k,t}$ be estimator for the 3ϵ-quantile of $\widehat{f}_{k,t}(\cdot)$ under $P_{k,t}$ such that $|\mathbf{Pr}_{X\sim P_{k,t}}[\widehat{f}_{k,t}(X) > \widehat{L}_{k,t}] - 3\epsilon| \leq 0.01\epsilon$, and then do $\widehat{L}_{k,t} \leftarrow \max\{\widehat{L}_{k,t}, \frac{0.1}{d}\widehat{\sigma}_{\mathrm{op}}\|v_{k,t}\|_2^2\}$.
    Let $\widehat{\tau}_{k,t}(x) = \widehat{f}_{k,t}(x)\mathbb{1}(\widehat{f}_{k,t}(x) > \widehat{L}_{k,t})$.
    Call SAMPLETOPEIGENVECTOR $(P, w_{k,t}, \epsilon, \gamma, 1/(k_{\mathrm{end}} \cdot t_{\mathrm{end}}))$. ▶{c.f. Algorithm 8.}
    Let $\widehat{\sigma}_{k,t}$ such that $|\widehat{\sigma}_{k,t} - \widehat{v}_{k,t}^\top\mathbf{\Sigma}\widehat{v}_{k,t}| \leq 4\gamma\widehat{v}_{k,t}^\top\mathbf{\Sigma}\widehat{v}_{k,t}$.
    (e.g., $\widehat{\sigma}_{k,t} := \mathbf{E}_{X\sim P}[w_{k,t}(X)\widehat{f}_{k,t}(X)\mathbb{1}(\widehat{f}_{k,t}(X) \leq \widehat{L}_{k,t})]$). ▶{c.f. Item 4 of Lemma 2.6.}
    Find an estimator $\widehat{\sigma}'_{k,t}$ such that $|\widehat{\sigma}'_{k,t} - \widehat{\sigma}_{k,t}| \leq 0.01\widehat{\sigma}_{k,t} + \frac{0.01\gamma}{r^2d}\|\widehat{v}_{k,t}\|_2^2\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}}$.
    Let $\widehat{T}_{k,t} = 2.35\gamma\widehat{\sigma}'_{k,t}$.
    $w_{k,t+1} \leftarrow \mathrm{HardThresholdingFilter}(P, w_{k,t}, \tau_{k,t}, \widehat{T}_{k,t}, R, \frac{0.1\gamma}{r^2d}\|\widehat{v}_{k,t}\|_2^2\|\mathbf{\Sigma}_{k,t})$, where $\widehat{\mathbf{\Sigma}}_{k,t}$ is a sample-based version of $\mathbf{\Sigma}_{k,t}$. ▶{c.f. Algorithm 4.}
  end for
  Set $w_{k+1,0} \leftarrow w_{k,t+1}$.
end for
return FAIL.

D.1. Establishing Condition D.3

In this section, we provide the estimators for Condition D.3, and prove the relevant guarantees.

D.1.1. ITEMS 1 AND 2

Items 1 and 2 of Condition D.3 rely on the fact that the empirical covariances used in the estimator $\prod_{\ell=1}^{p_k}\widehat{\mathbf{B}}_{k,t,\ell}$ are close enough to $\mathbf{B}_{k,t}$, as in Lemma D.4 below. We first prove the lemma and then show how the two conditions follow.

Lemma D.4. *Let $\delta < 1$. Consider Algorithm 6 and assume that $P_{k,t}$ is supported on an ℓ_2-ball of radius $r\sqrt{d\|\mathbf{\Sigma}\|_{\mathrm{op}}}$. If $\widehat{W}_{k,t}$ and every $\widehat{\mathbf{B}}_{k,t,\ell}$ in Algorithm 6 are calculated using*

$$\tilde{n} > C\frac{r^2dp^2}{\delta^2}\log\left(\frac{d}{\tau}\right)$$

samples, where C is a sufficiently large constant, then

$$\left\|\widehat{\mathbf{M}}_{k,t} - \mathbf{M}_{k,t}\right\|_2 \leq \delta\left\|\mathbf{M}_{k,t}\right\|_{\mathrm{op}},$$

with probability at least $1 - \tau$.

Proof. Let $\|\widehat{\mathbf{B}}_{k,t,\ell} - \mathbf{B}_{k,t}\|_{\mathrm{op}} \leq \delta_1\|\mathbf{B}_{k,t}\|_{\mathrm{op}}$ and $|\widehat{W}_{k,t} - \mathbf{E}_{X\sim P}[w_{k,t}(X)]| \leq \delta_2$, for δ_1, δ_2 to be determined. The accuracy of the estimators $\widehat{\mathbf{B}}_{k,t,\ell}$ and that of $\widehat{\mathbf{M}}_{k,t}$ are related as follows:

Algorithm 8 SAMPLETOPEIGENVECTOR 2

Input: Distribution P, weights w, parameters ϵ, γ, δ.
Let $\mathbf{M} := (\mathbf{E}_{X\sim P}[w(X)XX^\top])^p$ for $p = C\frac{\log(d/\gamma)}{\gamma}$.
$\widehat{r} \leftarrow 0$.
for $j \in [\log(1/\delta)]$ **do**
  Let $\widehat{\mathbf{M}}$ be a sample based estimator for $\mathbf{\Sigma}^p_{P_w}$ calculated using Algorithm 6.
  $y \leftarrow \widehat{\mathbf{M}}' \cdot g$ for $g \sim \mathcal{N}(0, \mathbf{I})$.
  $\widehat{r} \leftarrow \max(\widehat{r} \leftarrow, y^\top \widehat{\mathbf{\Sigma}}_{P_w} y/\|y\|_2^2)$.
end for ▶{cf. Lemma D.8}
Let $\widehat{\mathbf{M}}$ be a sample based estimator for $\mathbf{\Sigma}^p_{P_w}$ calculated using Algorithm 6.
$u \leftarrow \widehat{\mathbf{M}} z$ for $z \sim \mathcal{N}(0, \mathbf{I})$.
Let $\widehat{\sigma}_u$ be any approximation such that $|\widehat{\sigma}_u - u^\top \mathbf{\Sigma} u| \le 4\gamma u^\top \mathbf{\Sigma} u$, for example $\widehat{\sigma}_u := \mathbf{E}_{X\sim P}[w(X)(u^\top X)^2 \mathbb{1}((u^\top X)^2 \le Q)]$, where Q is within 0.01ϵ from the 3ϵ-quantile of $(u^\top X)^2$ under P_w.
Find estimator $\widehat{\sigma}'_u$ such that $|\widehat{\sigma}'_u - \widehat{\sigma}_u| \le 0.01\gamma\widehat{\sigma}_u + 0.01\gamma\|u\|_2^2\|\mathbf{\Sigma}\|_{\mathrm{op}}$.
Let $\widehat{\mathbf{\Sigma}}_{P_w}$ be a sample-based version of $\mathbf{\Sigma}_{P_w}$.
if $\widehat{\sigma}'_u \ge (1 - C\gamma)u^\top \widehat{\mathbf{\Sigma}}_{P_w} u$ and $\frac{u^\top \widehat{\mathbf{\Sigma}}_{k,t} u}{\|u\|_2^2} \ge (1-\gamma)\widehat{r}_t$ **then**
  return $u/\|u\|_2$. ▶{c.f. Lemma 2.7}
end if

Lemma D.5 (Lemma 4.14 in (Diakonikolas et al., 2022d)). *Let $\mathbf{A}, \mathbf{B}, \mathbf{B}_1, \ldots, \mathbf{B}_p$ be symmetric $d \times d$ matrices and define $\mathbf{M} = \mathbf{B}^p, \mathbf{M}_S = \prod_{i=1}^p \mathbf{B}_i$. If $\|\mathbf{B}_i - \mathbf{B}\|_2 \le \delta_1\|\mathbf{B}\|_2$, then $\|\mathbf{M}_S - \mathbf{B}^p\|_{\mathrm{op}} \le p\delta_1(1+\delta_1)^p\|\mathbf{B}\|_{\mathrm{op}}^p$.*

By Lemma D.5, it suffices to ensure $p\delta_1 e^{p\delta_1} < \delta$. For that, it suffices to use $\delta_1 := \delta/(3p)$. In the reminder we focus on ensuring $\|\widehat{\mathbf{B}}_{k,t,\ell} - \mathbf{B}_{k,t}\|_{\mathrm{op}} \le \delta_1\|\mathbf{B}_{k,t}\|_{\mathrm{op}}$ for that choice of δ_1.

Let $\|\widehat{\mathbf{\Sigma}}_{k,t,\ell} - \mathbf{\Sigma}_{k,t}\|_{\mathrm{op}} \le \delta_3\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}}$ for δ_3 to be specified. We have that

$$\begin{aligned}
\left\|\widehat{\mathbf{B}}_{k,t,\ell} - \mathbf{B}_{k,t}\right\|_{\mathrm{op}} &= \left\|\widehat{W}\widehat{\mathbf{\Sigma}}_{k,t,\ell} - \mathop{\mathbf{E}}_{X\sim P}[w_{k,t}(X)]\mathbf{\Sigma}_{k,t}\right\|_{\mathrm{op}} \\
&\le \mathop{\mathbf{E}}_{X\sim P}[w_{k,t}(X)]\left\|\widehat{\mathbf{\Sigma}}_{k,t,\ell} - \mathbf{\Sigma}_{k,t}\right\|_{\mathrm{op}} + \left|\widehat{W}_{k,t} - \mathop{\mathbf{E}}_{X\sim P}[w_{k,t}(X)]\right| \cdot \left\|\widehat{\mathbf{\Sigma}}_{k,t,\ell}\right\|_{\mathrm{op}} \\
&\le \left\|\widehat{\mathbf{\Sigma}}_{k,t,\ell} - \mathbf{\Sigma}_{k,t}\right\|_{\mathrm{op}} + \delta_2\left\|\widehat{\mathbf{\Sigma}}_{k,t,\ell}\right\|_{\mathrm{op}} \\
&\le (1+\delta_2)\left\|\widehat{\mathbf{\Sigma}}_{k,t,\ell} - \mathbf{\Sigma}_{k,t,\ell}\right\|_{\mathrm{op}} + \delta_2\|\mathbf{\Sigma}_{k,t,\ell}\|_{\mathrm{op}} \\
&\le (1+\delta_2)\delta_3\|\mathbf{\Sigma}_{k,t,\ell}\|_{\mathrm{op}} + \delta_2\|\mathbf{\Sigma}_{k,t,\ell}\|_{\mathrm{op}} \\
&\le 2\delta_3\|\mathbf{\Sigma}_{k,t,\ell}\|_{\mathrm{op}} + \delta_2\|\mathbf{\Sigma}_{k,t,\ell}\|_{\mathrm{op}} \ .
\end{aligned}$$

To make the RHS upper bounded by $\delta_1\|\mathbf{\Sigma}_{k,t,\ell}\|_{\mathrm{op}}$, we choose $\delta_3 = 0.1\delta_1$ and $\delta_2 =: 0.1\delta_1$. The sample complexity of achieving $|\widehat{W}_{k,t} - \mathbf{E}_{X\sim P}[w_{k,t}(X)]| \le \delta_2 = \delta/(30p)$ is $O(\frac{p^2}{\delta^2}\log(1/\tau))$ by Hoeffding bounds. The sample complexity for achieving $\|\widehat{\mathbf{\Sigma}}_{k,t} - \mathbf{\Sigma}_{k,t}\|_{\mathrm{op}} \le \delta_3\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}}$ is by Fact B.8:

$$\widetilde{n} \lesssim \frac{r^2 d\|\mathbf{\Sigma}\|_{\mathrm{op}}}{\delta_3^2\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}}}\log\left(\frac{d}{\tau}\right) \lesssim \frac{r^2 dp^2\|\mathbf{\Sigma}\|_{\mathrm{op}}}{\delta^2\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}}}\log\left(\frac{d}{\tau}\right) \lesssim \frac{r^2 dp^2}{\delta^2}\log\left(\frac{d}{\tau}\right) \ ,$$

where the last inequality uses that because of our assumptions $\mathbf{E}_{X\sim P}[w_{k,t}(X)] \ge 1 - O(\epsilon)$ and $\|\mathbf{B}_{k,t}\|_{\mathrm{op}} \ge 0.5\|\mathbf{\Sigma}\|_{\mathrm{op}}$, we have that $\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}} = \|\mathbf{B}_{k,t}\|_{\mathrm{op}}/\mathbf{E}_{X\sim P}[w_{k,t}(X)] \gtrsim \|\mathbf{\Sigma}\|_{\mathrm{op}}$.

■

Proof of Items 1 and 2 of Condition D.3. The two conditions follow directly from Lemma D.4 using $\delta = \frac{0.01}{\sqrt{d}}\min\left(\frac{\sqrt{\gamma/\epsilon}}{r}, 1\right)$. For the first one: Since $\|\widehat{\mathbf{M}}_{k,t} - \mathbf{M}_{k,t}\|_{\mathrm{op}} \le \frac{0.1}{r\sqrt{d}}\sqrt{\gamma/\epsilon}\|\mathbf{M}_{k,t}\|_{\mathrm{op}} \le \frac{0.1}{r\sqrt{d}}\sqrt{\gamma/\epsilon}\|\mathbf{M}_{k,t}\|_{\mathrm{F}}$ and we are

interested only for points with $\|x\|_2 \le r\sqrt{d\|\mathbf{\Sigma}\|_{\mathrm{op}}}$, we have that

$$g(x) \le \|\mathbf{M}_{k,t}x\|_2^2 \le 2\left\|\widehat{\mathbf{M}}_{k,t}x\right\|_2^2 + 2\left\|\left(\mathbf{M}_{k,t} - \widehat{\mathbf{M}}_{k,t}\right)x\right\|_2^2$$
$$\le 2\left\|\widehat{\mathbf{M}}_{k,t}x\right\|_2^2 + 2\|x\|_2^2\left\|\mathbf{M}_{k,t} - \widehat{\mathbf{M}}_{k,t}\right\|_{\mathrm{op}}^2 \le 2\widehat{g}(x) + 0.02(\gamma/\epsilon)\|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2\|\mathbf{\Sigma}\|_{\mathrm{op}} .$$

Equivalently, $\widehat{g}(x) \ge 0.5g(x) - 0.01(\gamma/\epsilon)\|\mathbf{M}_{k,t}\|_{\mathrm{F}}^2\|\mathbf{\Sigma}\|_{\mathrm{op}}$. The proof of Item 2 can be found in (Diakonikolas et al., 2022d). ■

D.1.2. ITEM 3

Lemma D.6. *For any distribution P over $\mathbb{R}$ and any $\epsilon, \delta \in (0,1)$, there is an estimator $\widehat{L}$ that uses $O\left(\frac{1}{\epsilon}\log\left(\frac{1}{\tau}\right)\right)$ samples from D and satisfies*

$$\left|\mathop{\mathbf{Pr}}_{X\sim P}[X > \widehat{L}] - \epsilon\right| \le \frac{\epsilon}{100} ,$$

with probability at least $1-\tau$. The memory usage and runtime of the estimator are also $O\left(\frac{1}{\epsilon}\log\left(\frac{1}{\tau}\right)\right)$.

Proof. The estimator $\widehat{L}(X_1,\dots,X_m)$ from m samples $X_1,\dots,X_m \sim P$ is defined to be the $(m\cdot\epsilon)$-greatest sample. Let L denote the target threshold, that is, the real for which $\mathbf{Pr}_{X\sim P}[X > L] = \epsilon$. Additionally, let $L_{-\epsilon}, L_{+\epsilon}$ such that $\mathbf{Pr}_{X\sim P}[L_- \le X \le L] = \mathbf{Pr}_{X\sim P}[L \le X \le L_+] = \epsilon/100$. Then, the probability that $\widehat{L}$ is not accurate is

$$\mathop{\mathbf{Pr}}_{X_1,\dots,X_m\sim P}\left[\left|\mathop{\mathbf{Pr}}_{X\sim P}[X > \widehat{L}] - \epsilon\right| > \frac{\epsilon}{100}\right] \le \mathbf{Pr}[\widehat{L} > L_+] + \mathbf{Pr}[\widehat{L} < L_-] .$$

Each term is bounded by an application of the multiplicative version of Chernoff bounds. Regarding first one, $\widehat{L} > L_+$ implies that we had at least $m\cdot\epsilon$ samples in the interval $(L_+, +\infty)$. But this is a low probability event because each sample belongs in this region with probability only $\epsilon(1-1/100)$:

$$\mathbf{Pr}[\widehat{L} > L_+] \le \mathop{\mathbf{Pr}}_{X_1,\dots,X_m\sim P}\left[\frac{1}{m}\sum_{j=1}^{m}\mathbb{1}(X_j > L_+) - \mathop{\mathbf{Pr}}_{X\sim P}[X > L_+] > \frac{\epsilon}{100}\right] \le e^{-\Omega(\epsilon m)} .$$

Choosing m to be a sufficiently large multiple of $\epsilon^{-1}\log\left(\frac{1}{\tau}\right)$ makes that probability of failure less than $\tau/2$. The other term $\mathbf{Pr}[\widehat{L} < L_-] < \tau/2$ can be shown similarly.

This algorithm stores $O(\epsilon^{-1}\log(1/\tau))$ one dimensional samples. If the algorithm selects the top $\epsilon\cdot m$ element by sorting the data first, its runtime is $O(\epsilon^{-1}\log(1/\tau)\log(\epsilon^{-1}\log(1/\tau)))$. However, $O(\epsilon^{-1}\log(1/\tau))$ is also possible for selection using a probabilistic divide and conquer algorithm. ■

D.2. Item 4

We prove Item 4 in the lemma below, which is a simple adaptation of Lemma 4.16 from Diakonikolas et al. (2022d). For the sake of completeness we include a proof.

Item 4 is used for estimating $\mathbf{E}_{X\sim P}[w(x)\tau(x)]$ in Line 3 of HARDTHRESHOLDINGFILTER as well as in lines 12 of Algorithm 8 and 21 of Algorithm 7. We briefly clarify how exactly it is used before giving the proof. We do this for Line 12 of Algorithm 8 since the other ones can be checked similarly: For that, we apply Lemma D.7 with $w(x) = w_{k,t}(x)\mathbb{1}((u^\top X)^2 \le Q)$. Then, the guarantee of the lemma is that (using the notation of Line 12) $|\widehat{\sigma}'_u - \widehat{\sigma}_u| \le 0.01\gamma\widehat{\sigma}_u + \frac{0.01\gamma}{r^2 d}\|u\|_2^2\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}} \le 0.01\gamma\widehat{\sigma}_u + 0.01\gamma\|u\|_2^2\|\mathbf{\Sigma}\|_{\mathrm{op}}$, where the last inequality follows from the fact that $\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}} \le r^2 d\|\mathbf{\Sigma}\|_{\mathrm{op}}$ as the support of $P_{k,t}$ is in a ball of radius $r\sqrt{d\|\mathbf{\Sigma}\|_{\mathrm{op}}}$ (c.f. Line 4 of Algorithm 7).

Lemma D.7. *In the setting of Algorithm 7, let r be a radius such that $\mathbf{Pr}_{X\sim G}[\|X\|_2 > r\sqrt{d\|\mathbf{\Sigma}\|_{\mathrm{op}}}] \le \epsilon$ and assume that $\|\mathbf{B}_{k,t}\|_{\mathrm{op}} \ge 0.5\|\mathbf{\Sigma}\|_{\mathrm{op}}$. For any τ and any weight function $w : \mathbb{R}^d \to [0,1]$ and any vector u, there is an estimator $\widehat{F}$ for the quantity $F_{k,t} := \mathbf{E}_{X\sim P}[w(X)(u^\top X)^2]$ that uses $n = O\left(\frac{r^4 d^2}{\gamma^2}\log(1/\tau)\right)$ samples from P, runs in $O(nd)$ time, uses memory $O(\log(1/\tau))$, and, with probability $1-\tau$ it satisfies $|\widehat{F} - F_{k,t}| \le 0.01\gamma F_{k,t} + \frac{0.01\gamma}{r^2 d}\|u\|_2^2\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}}$.*

Proof. We show the lemma for constant probability of success as then one can boost that to $1-\tau$ by repeating $\log(1/\tau)$ times and keeping the median. The estimator is just sample average. By Chebysev's inequality, with constant probability, we have that

$$\left|\frac{1}{n}\sum_{i=1}^{n} w(x_i)(u^\top x_i)^2 - \mathop{\mathbf{E}}_{X\sim P}[w(X)(u^\top X)^2]\right| \lesssim \frac{\sqrt{\mathbf{Var}_{X\sim P}[w(X)(u^\top X)^2]}}{n} .$$

We want to upper bound the RHS by $0.01\gamma\,\mathbf{E}_{X\sim P}[w(X)(u^\top X)^2] + \frac{0.01\gamma}{r^2 d}\|u\|_2^2\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}}$. A sufficient number of samples for that is big enough multiple of the following:

$$\begin{aligned}
\frac{\mathbf{Var}_{X\sim P}[w(X)(u^\top X)^2]}{\left(\gamma\,\mathbf{E}_{X\sim P}[w(X)(u^\top X)^2] + \frac{\gamma}{r^2 d}\|u\|_2^2\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}}\right)^2} &\le \frac{r^2 d\,\mathbf{Var}_{X\sim P}[w(X)(u^\top X)^2]}{\gamma^2\,\mathbf{E}_{X\sim P}[w(X)(u^\top X)^2]\|u\|_2^2\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}}} \\
&\le \frac{r^2 d\,\mathbf{E}_{X\sim P}[w^2(X)(u^\top X)^4]}{\gamma^2\,\mathbf{E}_{X\sim P}[w(X)(u^\top X)^2]\|u\|_2^2\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}}} \\
&\le \frac{r^2 d\|u\|_2^2 r^2 d\|\mathbf{\Sigma}\|_{\mathrm{op}}\,\mathbf{E}_{X\sim P}[w(X)(u^\top X)^2]}{\gamma^2\,\mathbf{E}_{X\sim P}[w(X)(u^\top X)^2]\|u\|_2^2\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}}} \\
&= \frac{r^4 d^2\|\mathbf{\Sigma}\|_{\mathrm{op}}}{\gamma^2\|\mathbf{\Sigma}_{k,t}\|_{\mathrm{op}}} \lesssim \frac{r^4 d^2}{\gamma^2} .
\end{aligned}$$

where the third line from the end used $w^2(x) \le w(x)$, $(u^\top x)^2 \le \|u\|_2^2\|x\|_2^2$, and $\|x\|_2^2 \le r^2 d\|\mathbf{\Sigma}\|_{\mathrm{op}}$ (by the pruning of Lines 3 and 4 in Algorithm 7) and the fourth line uses our assumption $\|\mathbf{B}_{k,t}\|_{\mathrm{op}} \ge 0.5\|\mathbf{\Sigma}\|_{\mathrm{op}}$. ■

D.2.1. ITEMS 5 AND 6 (POWER ITERATION)

Regarding Item 6 we use the lemma below with $\mathbf{A} = \mathbf{B}_{k,t}$ and $\widehat{\mathbf{A}} = \widehat{\mathbf{M}}_{k,t}$, for which it is guaranteed by Lemma D.4 that $\|\widehat{\mathbf{A}} - \mathbf{A}^p\|_{\mathrm{op}} \le \delta\|\mathbf{A}^p\|_{\mathrm{op}} \le \delta\|\mathbf{A}^p\|_{\mathrm{F}}$.

Item 5 of Condition D.3 follows as a corollary of the same lemma, since we can boost the probability of success from 0.9 to arbitrarily close to 1 by repeating the procedure and returning the vector y maximizing $y^\top\mathbf{A}y/\|y\|_2^2$.

Lemma D.8. *Let $\delta > 0$, $\gamma \in (0, 1/2)$, $p \in \mathbb{N}$ and $d \times d$ PSD matrices $\mathbf{A}, \widehat{\mathbf{A}}$, such that $\|\widehat{\mathbf{A}} - \mathbf{A}^p\|_{\mathrm{op}} \le \delta\|\mathbf{A}^p\|_{\mathrm{F}}$. Let $z \sim \mathcal{N}(0, \mathbf{I})$ and $y = \widehat{\mathbf{A}}z$. Then, if $p > C\log(d/\gamma)/\gamma$ for a sufficiently large constant, and $\delta < c\gamma/\sqrt{d}$ for sufficiently small positive constant, the following holds with probability at least 0.9:*

$$\frac{y^\top\mathbf{A}y}{\|y\|_2^2} \ge (1 - O(\gamma))\|\mathbf{A}\|_{\mathrm{op}} .$$

Proof. Denote $\mathbf{\Delta} := \widehat{\mathbf{A}} - \mathbf{A}^p$, which has operator norm at most $\delta\|\mathbf{A}^p\|_{\mathrm{F}}$. We first analyze the random variable $X = z^\top(\mathbf{A}^p\mathbf{\Delta} + \mathbf{\Delta}\mathbf{A}^p)z$. Then

$$\mathbf{E}[X] = \mathrm{tr}\,(\mathbf{A}^p\mathbf{\Delta} + \mathbf{\Delta}\mathbf{A}^p) = 2\langle\mathbf{A}^p, \mathbf{\Delta}\rangle \le 2\|\mathbf{A}^p\|_{\mathrm{F}}\|\mathbf{\Delta}\|_{\mathrm{F}} \le 2\sqrt{d}\|\mathbf{A}^p\|_{\mathrm{F}}\|\mathbf{\Delta}\|_{\mathrm{op}} \le 2\delta\sqrt{d}\|\mathbf{A}^p\|_{\mathrm{F}}^2,$$

since $\|\mathbf{\Delta}\|_{\mathrm{op}} \le \delta\|\mathbf{A}\|_{\mathrm{F}}$. And since $\mathbf{A}^p\mathbf{\Delta} + \mathbf{\Delta}\mathbf{A}^p$ is symmetric, by Fact 2.2, the variance of X satisfies $\mathbf{Var}(X) \lesssim \|\mathbf{A}^p\mathbf{\Delta} + \mathbf{\Delta}\mathbf{A}^p\|_{\mathrm{F}}^2 \lesssim \|\mathbf{\Delta}\|_{\mathrm{op}}^2\|\mathbf{A}^p\|_{\mathrm{F}}^2 \le \delta^2\|\mathbf{A}^p\|_{\mathrm{F}}^4$. Therefore, w.h.p,

$$|z^\top(\mathbf{A}^p\mathbf{\Delta} + \mathbf{\Delta}\mathbf{A}^p)z| \lesssim \delta\sqrt{d}\|\mathbf{A}^p\|_{\mathrm{F}}^2.$$

By a similar argument, we obtain that with high probability,

$$|z^\top\left(\mathbf{A}^{p+1}\mathbf{\Delta} + \mathbf{\Delta}\mathbf{A}^{p+1}\right)z| \lesssim \delta\sqrt{d}\|\mathbf{A}\|_{\mathrm{op}}\|\mathbf{A}^p\|_{\mathrm{F}}^2.$$

Now let $X' = z^\top\mathbf{A}^{2p}z$. Then $\mathbf{E}[X'] = \mathrm{tr}(\mathbf{A}^{2p}) = \|\mathbf{A}^p\|_{\mathrm{F}}^2$. By Gaussian anticoncentration (Fact 2.2), with high probability:

$$z^\top\mathbf{A}^{2p}z \asymp \|\mathbf{A}^p\|_{\mathrm{F}}^2 . \tag{47}$$

Also, $\|z\|_2^2 \lesssim d$. We now have the following:

$$\begin{aligned}
\|y\|_2^2 &= z^\top \widehat{\mathbf{A}}^2 z \\
&= z^\top \left(\mathbf{A}^p + \mathbf{\Delta}\right)^2 z \\
&= z^\top \mathbf{A}^{2p} z + z^\top \left(\mathbf{A}^p \mathbf{\Delta} + \mathbf{\Delta}\mathbf{A}^p\right) z + z^\top \mathbf{\Delta}^2 z \\
&\le z^\top \mathbf{A}^{2p} z + C\delta\sqrt{d}\|\mathbf{A}^p\|_{\mathrm{F}}^2 + Cd\|\mathbf{\Delta}\|_{\mathrm{op}}^2 \\
&\le z^\top \mathbf{A}^{2p} z + C\|\mathbf{A}^p\|_{\mathrm{F}}^2 \left(\delta\sqrt{d} + d\delta^2\right) \\
&\le z^\top \mathbf{A}^{2p} z + 2C\delta\sqrt{d}\|\mathbf{A}^p\|_{\mathrm{F}}^2 \,. \qquad (48)
\end{aligned}$$

Similarly,

$$\|y\|_2^2 \ge z^\top \mathbf{A}^{2p} z - C\delta\sqrt{d}\|\mathbf{A}^p\|_{\mathrm{F}}^2 \,. \qquad (49)$$

By Fact 2.3, we also know that with probability 0.99 we have that

$$\frac{z^\top \mathbf{A}^{2p+1} z}{z^\top \mathbf{A}^{2p} z} \ge (1-\gamma)\|\mathbf{A}\|_{\mathrm{op}} \,. \qquad (50)$$

Conditioned on this event, we have that:

$$\begin{aligned}
y^\top \mathbf{A} y &= z^\top \widehat{\mathbf{A}}\mathbf{A}\widehat{\mathbf{A}} z \\
&= z^\top \left(\mathbf{A}^p + \mathbf{\Delta}\right)\mathbf{A}\left(\mathbf{A}^p + \mathbf{\Delta}\right) z \\
&= z^\top \mathbf{A}^{2p+1} z + z^\top (\mathbf{A}^{p+1}\mathbf{\Delta} + \mathbf{\Delta}\mathbf{A}^{p+1}) z + z^\top (\mathbf{\Delta}\mathbf{A}\mathbf{\Delta}) z \\
&\ge z^\top \mathbf{A}^{2p+1} z - C\delta\sqrt{d}\|\mathbf{A}\|_{\mathrm{op}}\|\mathbf{A}^p\|_{\mathrm{F}}^2 \\
&\ge (1-\gamma)\|\mathbf{A}\|_{\mathrm{op}} z^\top \mathbf{A}^{2p} z - C\delta\sqrt{d}\|\mathbf{A}\|_{\mathrm{op}}\|\mathbf{A}^p\|_{\mathrm{F}}^2 && \text{(using (50))} \\
&\ge (1-\gamma)\|\mathbf{A}\|_{\mathrm{op}} \left(z^\top \mathbf{A}^{2p} z - 3C\delta\sqrt{d}\|\mathbf{A}^p\|_{\mathrm{F}}^2\right) && (\gamma < 2/3) \\
&\ge (1-\gamma)\|\mathbf{A}\|_{\mathrm{op}} \left(\|y\|^2 - 5C\delta\sqrt{d}\|\mathbf{A}^p\|_{\mathrm{F}}^2\right) \,. && \text{(using (48))}
\end{aligned}$$

Finally, dividing by $\|y\|_2^2$ both sides yields

$$\frac{y^\top \mathbf{A} y}{\|y\|_2^2} \ge (1-\gamma)\|\mathbf{A}\|_{\mathrm{op}} \left(1 - 5C\delta\sqrt{d}\|\mathbf{A}^p\|_{\mathrm{F}}^2 \frac{1}{\|y\|_2^2}\right).$$

We will now show that

$$5C\delta\sqrt{d}\|\mathbf{A}^p\|_{\mathrm{F}}^2 \frac{1}{\|y\|_2^2} \lesssim \gamma,$$

which will complete the result.

$$\begin{aligned}
\delta\sqrt{d}\|\mathbf{A}^p\|_{\mathrm{F}}^2 \frac{1}{\|y\|_2^2} &\le \frac{\delta\sqrt{d}\|\mathbf{A}^p\|_{\mathrm{F}}^2}{z^\top \mathbf{A}^{2p} z - 2C\delta\sqrt{d}\|\mathbf{A}^p\|_{\mathrm{F}}^2} && \text{(using (49))} \\
&\lesssim \frac{\delta\sqrt{d}\|\mathbf{A}^p\|_{\mathrm{F}}^2}{\|\mathbf{A}^p\|_{\mathrm{F}}^2 - C'\delta\sqrt{d}\|\mathbf{A}^p\|_{\mathrm{F}}^2} && \text{(using (47))} \\
&= \frac{\delta\sqrt{d}}{1 - C'\delta\sqrt{d}} \\
&\lesssim \delta\sqrt{d} \lesssim \gamma. && \text{(using } \delta < c\gamma/\sqrt{d})
\end{aligned}$$

■

D.3. Proof sketch of Theorem D.2

The proof of correctness of Algorithm 7 is the same as in Appendix C modulo small changes in the constants of the bounds to account for the usage of our sample-based estimators. Thus, here we only derive the sample complexity, memory, and runtime bounds.

Sample complexity: Consider the (k,t)-th iteration of Algorithm 7 (that is, the k-th iteration of the outer loop and t-th iteration of the inner loop). In order to ensure all parts of Condition D.3 we use Lemma D.4 with $\delta = \frac{0.01}{\sqrt{d}} \min\left(\frac{\sqrt{\gamma/\epsilon}}{r}, \gamma\right)$ and probability of failure τ a sufficiently large polynomial of d/ϵ so that with high probability the estimators are accurate across all iterations simultaneously. Thus, each estimator $\widehat{\mathbf{B}}_{k,t,\ell}$ uses $(r^2 p_k^2 d/\delta^2)\mathrm{polylog}(d/\epsilon)$ samples. We use p_k of these estimators in Line 13 resulting in $(r^2 p_k^3 d/\delta^2)\mathrm{polylog}(d/\epsilon)$ samples. Moreoever, for the estimator of Line 16 we use Lemma D.6 (again with $\tau = \mathrm{poly}(d/\epsilon)$) which contributes another $(1/\epsilon)\mathrm{polylog}(d/\epsilon)$ to the sample complexity. We also call the estimator of Lemma D.7 which uses $O((r^4 d^2/\gamma^2)\mathrm{polylog}(d/\epsilon))$ many samples. Thus, the number of samples used during the (k,t)-th iteration of Algorithm 7 is $n_{k,t} := O\left(r^2 p_k^3 d/\delta^2 \mathrm{polylog}(d/\epsilon) + (1/\epsilon)\mathrm{polylog}(d/\epsilon) + (r^4 d^2/\gamma^2)\mathrm{polylog}(d/\epsilon)\right)$, and the total sample complexity is

$$\begin{aligned} n &= \sum_{k=1}^{k_{\mathrm{end}}} \sum_{t=1}^{t_{\mathrm{end}}} n_{k,t} \lesssim t_{\mathrm{end}} \left(\sum_{k=1}^{k_{\mathrm{end}}} p_k^3\right) \frac{r^2 d}{\delta^2}\mathrm{polylog}(d/\epsilon) + \frac{t_{\mathrm{end}} k_{\mathrm{end}}}{\epsilon}\mathrm{polylog}(d/\epsilon) + t_{\mathrm{end}} k_{\mathrm{end}} \frac{r^4 d^2}{\gamma^2}\mathrm{polylog}(d/\epsilon) \\ &\lesssim \frac{r^2 d^2 \max(r^2\epsilon, 1)}{\gamma^5}\mathrm{polylog}(d/\epsilon) + \frac{1}{\epsilon\gamma}\mathrm{polylog}(d/\epsilon) + \frac{r^4 d^2}{\gamma^3}\mathrm{polylog}(d/\epsilon) \end{aligned}$$

Memory Usage: The estimator of Line 16 in Algorithm 7 uses memory $O((1/\epsilon)\mathrm{polylog}(d/\epsilon))$, and that memory can be freed and reused the next time Line 16 is executed. The filters created by HARDTHRESHOLDINGFILTER are of the form of a vector and a one dimensional threshold, meaning that they can be stored in $O(d)$ memory. The number of the filters created in each one of the $t_{\mathrm{end}} \cdot k_{\mathrm{end}}$ calls of HARDTHRESHOLDINGFILTER is at most $O(\log(d/\epsilon))$. Thus, the memory used to store all of them is $O(t_{\mathrm{end}} \cdot k_{\mathrm{end}} \cdot d \log(d/\epsilon)) = O((d/\gamma)\mathrm{polylog}(d/\epsilon))$. Every other operation done in Algorithm 7 is either a multiplication of an empirical second moment matrix with a vector (that as we have noted earlier can be implemented in $O(d)$ memory) or even more basic operation.

Runtime: The same runtime analysis from Appendix C.3.1 applies.

D.3.1. BIT COMPLEXITY

Bit Complexity We briefly discuss how our algorithm can be implemented with bounded precision (i.e., in the standard word RAM model) using $(d/\gamma)\mathrm{polylog}(d/\epsilon)$ registers of size $(1/\gamma)\mathrm{polylog}(d/\epsilon)$ bits each. We assume that all but ϵ mass of the inlier distribution is supported in a ball of radius $R = (d/\epsilon)^{\mathrm{polylog}(d/\epsilon)}$ and we also assume that $\mathbf{\Sigma} \succeq (\epsilon/d)^{\mathrm{polylog}(d/\epsilon)}\mathbf{I}$. Our algorithm ignores all points x outside of the ball and rounds the remaining ones to x' such that $\|x - x'\|_2 \le \eta$. We want η to be small enough so that it does not affect the correctness of our algorithm. Picking $\eta = (\epsilon/d)^{\mathrm{polylog}(d/\epsilon)}$ means that the rounded distribution has covariance matrix close enough to the original one so that is satisfies the same stability condition (modulo a difference in the constants), thus our algorithm will be correct. So far, by the aforementioned rounding, every point can be stored in d words of size $O(\log(Rd/\eta)) = \mathrm{polylog}(d/\epsilon)$ bits). Regarding the bit complexity of intermediate calculations of our algorithm, the most expensive one is the computation of $\mathbf{M}z$ for $\mathbf{M}$ being the empirical covariance raised to the power $p = \gamma^{-1}\mathrm{polylog}(d/\epsilon)$ and z a random Gaussian vector. Since the power p includes a $1/\gamma$ factor, the final result may have bit complexity increased by $1/\gamma$ factor, thus we need d registers of size $(1/\gamma)\mathrm{polylog}(d/\epsilon)$ to store the resulting vector $\mathbf{M}z$. The filters that we create are $(1/\gamma)\mathrm{polylog}(d/\epsilon)$ in number and the representation of each is a vector of the previous form (and a one-dimensional threshold). Thus, overall, we have to use $(d/\gamma)\mathrm{polylog}(d/\epsilon)$ many registers of size $(1/\gamma)\mathrm{polylog}(d/\epsilon)$ bits each.